A Reputation-based Framework for Honest Provenance Reporting

LINA BARAKAT, University of Essex, UK
PHILLIP TAYLOR and NATHAN GRIFFITHS, University of Warwick, UK
SIMON MILES, King's College London, UK

Given the distributed, heterogenous, and dynamic nature of service-based IoT systems, capturing circumstances data underlying service provisions becomes increasingly important for understanding process flow and tracing how outputs came about, thus enabling clients to make more informed decisions regarding future interaction partners. Whilst service providers are the main source of such circumstances data, they may often be reluctant to release it, e.g. due to the cost and effort required, or to protect their interests. In response, this paper introduces a reputation-based framework, guided by intelligent software agents, to support the sharing of truthful circumstances information by providers. In this framework, *assessor agents*, acting on behalf of clients, rank and select service providers according to reputation, while *provider agents*, acting on behalf of service providers, learn from the environment and adjust provider's circumstances provision policies in the direction that increases provider profit with respect to perceived reputation. The novelty of the reputation assessment model adopted by assessor agents lies in affecting provider reputation scores by whether or not they reveal truthful circumstances data underlying their service provisions, in addition to other factors commonly adopted by existing reputation schemes. The effectiveness of the proposed framework is demonstrated through an agent-based simulation including robustness against a number of attacks, with a comparative performance analysis against FIRE as a baseline reputation model.

CCS Concepts: • **Computing methodologies** → **Knowledge representation and reasoning**; **Intelligent agents**.

Additional Key Words and Phrases: Reputation, Circumstances, Honest Reporting, Provenance

ACM Reference Format:
Lina Barakat, Phillip Taylor, Nathan Griffiths, and Simon Miles. 2021. A Reputation-based Framework for Honest Provenance Reporting. *ACM Trans. Internet Technol.* 1, 1, Article 1 (January 2021), 30 pages. https://doi.org/10.1145/3507908

1 INTRODUCTION

The advances in computing, communication and information technologies have enabled the emergence of the Internet of Things (IoT), where a growing number of physical objects (or *things*) are connected to the virtual world, interacting and coordinating with each other in seamless manner, to achieve sophisticated tasks for end users [53]. Such connectivity brings many advantages to organisations, businesses, societies and individuals, and has impacted a number of application domains such as smart cities, smart transportation, e-health, etc [1]. Service-orientation is a particularly promising paradigm for IoT-based systems [18, 24, 56]. *Services* are self-contained and platform-independent reusable computational entities that are described, published, discovered and invoked over the network using accepted standard languages and protocols. Via abstracting networked objects (e.g., physical devices, information resources, or functionalities) as services,

Authors' addresses: Lina Barakat, University of Essex, UK, lina.barakat@essex.ac.uk; Phillip Taylor, philip.m.taylor@warwick.ac.uk; Nathan Griffiths, nathan.griffiths@warwick.ac.uk, University of Warwick, UK; Simon Miles, simon.miles@kcl.ac.uk, King's College London, UK.

and utilising these services as elementary building blocks, this paradigm supports the rapid and economic development of complex, interoperable distributed applications. However, such systems exhibit high degrees of heterogeneity, dynamism, and uncertainty, due to distribution, participant autonomy and lack of local control.

Under such heterogeneous, dynamic and uncertain settings, availability of context and circumstances data (in interoperable format) becomes particularly important for understanding the processes under which service provisions took place, tracing the context of achieved result data, and providing clients with useful information to support their decision making in selecting a future service provider. In particular, when clients have a rich description of the circumstances in which services were provided, they are able to judge to what degree such circumstances match those of interest, thus making more informed provider selection decisions [17, 34]. For example, the operation of a sensor-based service may be affected by a one-off freak event (e.g. ash from an erupting volcano, flood, etc), leading to inaccurate readings and poor service, but for which the provider should not be blamed as it is out of their control. Similarly, reliance on sub-providers may have an effect on the provider's performance, with the provider potentially operating differently with different sub-contractors. The PROV standard [58] (published by W3C as a standard for interoperable provenance) provides a suitable solution for generating (and interpreting) circumstances information by system members. A PROV document describes in a queryable form the causes and effects within a particular past process of a system (such as agents interacting, the execution of a program, or enactment of a physical world process), as a directed graph with annotations. The contents of a provenance graph can be collated from data recorded by a set of independent agents, and clients have a standard means to query the data, e.g. by SPARQL[1] queries.

Service providers are the obvious source of such provenance data, as it is a record of how they provided a service. However, being autonomous and self-interested, providers may not be willing to release such records for several reasons. This may be due to the additional burden incurred on the provider side (the process of provenance recording could be expensive), or for competitive grounds (e.g. it may be against provider interests to release records showing that they performed poorly). Providers may even claim untrue events (e.g. falsely claiming out-of-control circumstances) to justify a poor performance. Therefore, providing relevant *incentives* to providers is a promising way to encourage them to release provenance data. In fact, various economic and psychological studies emphasise the importance of incentives in directing an entity towards a desired behaviour [12, 15]. Generally, such incentives could be intrinsic (e.g. out of personal interest in the task) or extrinsic (e.g. to gain financial reward or social approval). In the context of service-oriented marketplaces, *reputation* is a particularly attractive (extrinsic) incentive for service providers. This is because clients in such systems rely on third-party providers to execute services on their behalf, and thus very commonly utilise provider reputation as an effective measure to assess the degree of risk prior to such reliance. Given this, a provider's reputation has a direct effect on its selection chances by clients, and consequently on its profit, etc.

In response, this paper proposes an agent-oriented middleware that supports and encourages the sharing of *truthful* circumstances reports by service providers (see Figure 1). Two types of software agents guide the middleware: *provider agents* that act on behalf of service providers, taking rational decisions with respect to provision of circumstances, and *assessor agents* that act on behalf of service clients, performing reputation assessments of providers that take into consideration providers's circumstances provision behaviour. In particular, the paper makes the following contributions.

- *Verification Mechanism of Circumstances Information.* Based on the provenance patterns for corroboration proposed in [2], we provide a mechanism for an assessor agent to estimate the *reliability* of a PROV circumstances claim from a provider, via confirming it against the claims of relevant witnesses.

[1] http://www.w3.org/TR/sparql11-overview/

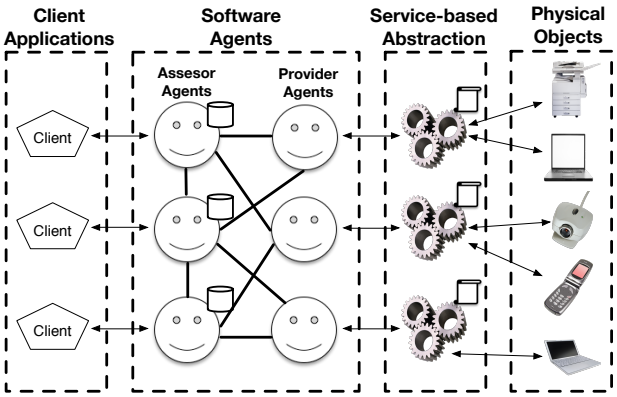


Fig. 1. Agent-Augmented Service Marketplace

- *Circumstance-aware Reputation Model.* We extend existing reputation models by incorporating the *availability* and *reliability* of circumstances data underlying service provisions into the reputation assessment process. We propose to do so by affecting two reputation-related factors: the *weights* of past service ratings, governing relevance of ratings for reputation assessment based on contextual circumstances; and the *confidence* in such weights, affecting in turn the overall confidence in the reputation score. We illustrate how these circumstance-related extensions can be injected into a number of existing reputation mechanisms, and assess their effectiveness in the simulation model via analysing performance against FIRE [23] as an example baseline model.
- *Provider Behaviour Model and Adaptation Mechanism.* We provide a behaviour model for provider agents, guiding their actions with respect to provision of various types of circumstances. The model is equipped with a rational learning algorithm, inspired by Policy Hill-Climbing [7], allowing a provider agent to find a policy that maximises their future utilities with respect to perceived reputation (in terms of the number of client requests received).

The remainder of the paper is organised as follows. Section 2 introduces the background concepts and related work. An abstraction of existing reputation models is presented in Section 3, followed by the proposed framework in Section 4. Our simulation model enabling the framework evaluation and the corresponding experimental results are presented in Section 5 and Section 6, respectively. Finally, Section 7 concludes the paper.

2 BACKGROUND

2.1 Reputation

Trust and reputation are concepts commonly modelled in mechanisms for improving the success of interactions by minimising uncertainty when self-interested individuals interact [43]. Trust is an assessment of the likelihood that an individual or organisation will cooperate and fulfil its commitments [14], while reputation can be viewed as the public perception of the trustworthiness of a given entity [27]. Given the importance of reputation in real-world environments, there continues to be active research interest in the area. Several computational models of trust and reputation exist (see [16, 26, 47] for comprehensive reviews), which can be broadly categorised into those focused on trusting identity (credential-based approaches) and those focused on trusting behaviour (experience based approaches). Credential-based approaches use policies to express when, for what, and how to determine trust based on certificates, keys, or digital

signatures, etc. Although such methods are effective for managing access, rights, and permissions, they do not support more general reasoning about interactions, and therefore in this paper we focus on experience based approaches, which are studied by researchers from many domains.

In multi-agent systems, most established computational reputation models, such as the Beta Reputation System (BRS) [25], TRAVOS [55], BLADE [44], HABIT [54], ReGreT [46, 48], and FIRE [23] use a combination of direct and indirect experience to derive a numerical or probabilistic assessment of reputation [59]. BRS [25] takes a probabilistic approach to assessing trust, with the outcomes of interactions recorded as binary variables from which the expected value of success of future interactions is estimated using a beta probability density function. If the confidence level of the estimate is below some predetermined threshold then the opinions of others are sought to inform the assessment and increase confidence. TRAVOS [55] builds on BRS, but accounts for untrustworthy witnesses by decreasing their impact. The use of a binary variable (success or failure) to model outcomes is a limitation of TRAVOS and alternative approaches have been proposed. For example, BLADE [44] models agents and advisor evaluation functions as dynamic random variables using Dirichlet distributions, enabling progressive learning of probabilistic models through Bayesian techniques. HABIT [54] records the context of outcomes, and uses a probabilistic approach, creating a Bayesian network to support reasoning about reputation. However, HABIT assumes that the distribution of an agent's behaviour is static, an assumption not made by other approaches. ReGreT [46, 48] assesses reputation on three aspects: (i) an individual dimension from direct experience, (ii) a social dimension using knowledge of others' experiences and the social structure, and (iii) an ontological dimension that accounts for different reputation-informing aspects (e.g. delivery, price, quality). FIRE [23] combines four different types of reputation and trust: interaction trust from direct experience, witness reputation from third party reports, role-based trust, and certified reputation based on third-party references [23]. The direct experience and witness reputation components are based on ReGreT.

Trust and reputation models have also been developed and applied in service-oriented systems. For example, Maximilien et al. [33] estimate the reputation of a service with respect to a quality by aggregating the previously observed quality values for this service (shared and accessible to all assessors). Similarly, Xu et al. [62] extend the UDDI registry with a reputation manager, aggregating the past ratings of a service into a reputation score. Malik et al. [32] propose a decentralised approach for service reputation assessment, where customers seek ratings from their peers, with the credibility of ratings being estimated based on deviation from the majority opinion.

More recently, a number of architectures and models have been proposed for managing and computing trust and reputation in IoT environments [13, 19], to guide interactions and filter out malicious nodes. These can be categorised into either centralised [4, 5, 10, 38] or distributed [9, 41]. In the centralised systems, feedback on previous interactions with IoT entities are reported to and managed by a central authority, which aggregates them and provides trust and reputation information. In the decentralised systems, trust and reputation management is performed by IoT-entities, which propagate their experiences and opinions to each other and aggregate them along with their own. In both centralised and decentralised systems, several factors affecting trust and reputation assessments have been considered, utilising techniques similar to those developed for multi-agent systems. This includes accounting for direct and indirect experiences with an entity [3, 9, 40, 41], social relationships among entity owners [9, 40, 41], and context [4, 5].

Existing experience-based reputation assessment models differ in their implementation details, but there are a number of characteristics that are usually common among them. An abstraction of these characteristics is presented in Section 3, upon which we propose our incentivisation-driven extension. In particular, existing approaches focus on the client's observation of the ultimate outcome of a service provision, without accounting for the availability of circumstances behind such provision. For example, circumstances underlying a delayed/failed delivery may include a freak event such

as flooding, or an incompetent sub-contractor failing to deliver on time. Without such circumstances, potentially relevant information is omitted from assessment, such as reasons for past failures, past alliances or affiliations, or changes in environment. Our model accounts for this limitation via affecting the reputation of a service provider by whether or not they reveal provision circumstances and the honesty of revealed circumstances. Such circumstances-related effects are incorporated as extensions to existing reputation models by influencing existing factors, as illustrated in Section 4.

2.2 Provenance

In order to make rich decisions, data about what has occurred in the past is required. This cannot be simply a collection of logged events, but must express the causality between them, otherwise it is impossible to discern which agent's actions led to the success or failure of service provision. Thus, we need the facilities to record and later access the provenance of such results, i.e. how they have come into being through processes and interactions. The amount of literature on provenance technologies has exploded in recent years, including surveys of the field [35, 52]. Key issues regarding provenance of data in distributed systems (such as service marketplaces) include how to model, record, store and query the provenance data; and how to adapt systems so that they become *provenance-aware*. Over the past decade, attempts have been made to unify the approaches taken in different research communities. This led to the development of a common model for provenance, the Open Provenance Model (OPM) [36], already widely used in academic and industrial projects. The W3C then initiated efforts to produce a standard for provenance modelling and access on the web, drawing on the OPM as well as many other relevant initiatives, such as Dublin Core. This effort concluded with a W3C recommendation, PROV [37, 58], which specifies the model, its semantics, and its serialisation for semantic web applications. PROV is adopted in this paper.

A PROV document describes in a queryable form the causes and effects within a particular past process of a system as a directed graph with annotations. In a PROV graph, an *activity* is something that has taken place, making *use* of or *generating entities*, such as data, physical or other things. *Agents* are parties responsible for (*associated with*) activities taking place. Activities, entities and agents (graph nodes) may be annotated with key-value *attributes* describing features that the elements had. *Timestamps* can also be added to show when entities were used or generated by activities.

2.3 Incentivisation

Various economic and psychological studies emphasise the importance of incentives in directing an entity towards a desired behaviour [12, 15]. Such studies reveal that there is typically a mixture of motives driving an entity to undertake a particular task, which could be intrinsic (e.g. out of personal interest in the task) or extrinsic (e.g. to gain financial reward or social approval), and may differ among entities. Targeting such motives with relevant incentives would thus allow pushing the entity's behaviour in the desired direction. The applicability of such incentivisation in computational systems has been investigated by a number of researchers. In particular, some work here has focused on studying the effect of punishment (in the form of fines) as a monetary incentivisation to establish a desired behaviour among a population of self-interested agents [21, 31, 39, 49]. Researchers have also considered other forms of incentivisation to promote cooperation among agents, including credibility scores [42], violation alerts [57], and deferred reciprocity [45].

Reputation based incentivisation has also been adopted to promote a particular behaviour [20]. For example, in order to motivate buyers to provide truthful ratings for sellers, Zhang et al [63] propose a trust-based incentive mechanism, where the reputations of buyers are modelled, and utilised by sellers to decide on the quality of the products offered to buyers. In particular, honest buyers are likely to become reputable due to being favoured by other buyers to form a social network, and respectively would benefit from better product offers from sellers. Our work also utilises reputation to

encourage agents to report the circumstances in which their services were provided. However, rather than introducing the reputation related to circumstances provision as an additional measure, the proposed framework aims to influence an existing reputation-driven system, where a particular reputation mechanism is already in place by clients to select between alternative providers. For this purpose, we introduce relevant circumstances-related factors, and embed these factors into the existing reputation mechanism. Since circumstances are not normally supplied by providers, the factors introduced encourage both the provision of information and the truthfulness of this information.

3 REPUTATION MODEL ABSTRACTION

A generic abstraction of the rating model commonly used by existing reputation approaches in the literature, is a tuple:

$$\langle P, A, I(a,p), r(i,q), ctx(i) \rangle \tag{1}$$

Here, P is the set of all service providers in the marketplace. A is the set of all assessors (clients) in the marketplace. Note that sets P and A are not necessarily distinct (e.g. an agent can act as both a client and a provider). $I(a,p)$ is the set of interactions that occurred between client $a \in A$ and provider $p \in P$. $r(i,q)$ is the rating that assessor a assigned to provider p for term q (e.g. for quality, timeliness, etc.), in their previous interaction $i \in I(a,p)$. For an overall perspective on the interaction, q = overall. Finally, $ctx(i)$ is the set of contextual circumstances under which interaction i occurred. A commonly used context is the point in time, $t(i)$, at which interaction i took place.

Based on this rating model, a generic abstraction of an experience-based reputation assessment model, which we also refer to as the *base reputation model*, can be summarised as a tuple (example instantiations of this base model, illustrating different reputation approaches in the literature, are presented in the appendix):

$$\langle \mathcal{W}_b(a,p,i), \mathcal{T}_b(a,p,q), \mathcal{F}_b(a,p,q) \rangle \tag{2}$$

Here, $\mathcal{W}_b(a,p,i)$ is a weighting factor governing the relevance of interaction i, with consideration of its contextual circumstances $ctx(i)$, for the assessment of provider p by assessor a. $\mathcal{T}_b(a,p,q)$ is the reputation score that assessor a assigns to provider p on term q. It is usually estimated by applying some summary function, $\bigoplus$, over the ratings of interactions $i \in I(\_,p)$ (where _ matches any value), while accounting for their weights $\mathcal{W}_b(a,p,i)$, i.e.

$$\mathcal{T}_b(a,p,q) = \bigoplus\nolimits_{i \in I(\_,p)} \langle \mathcal{W}_b(a,p,i), r(i,q) \rangle \tag{3}$$

For example, $\bigoplus$ may correspond to a weighted mean over numeric ratings, a probability estimation measure over binary ratings, or some other kind of function. $\mathcal{F}_b(a,p,q)$ is the degree of confidence in the computed reputation score $\mathcal{T}_b(a,p,q)$. It may depend on various factors, including the number of interactions upon which the reputation is assessed. For example, a high confidence $\mathcal{F}_b$ in a low reputation score $\mathcal{T}_b$ might mean that there is a large number of past observations supporting the judgment that the provider performs poorly.

4 INCENTIVISATION FRAMEWORK

To influence a provider towards provision of (true) circumstances reports, the incentivisation mechanism in place should allow the provider to gain some utility in response. In this paper, we follow a reputation-based approach for incentivisation, where providers achieve reputation gains when revealing the desired information. The basic principle underpinning our approach is that clients need to select trustworthy partners, and providers aim to be trusted, and so can be incentivised via reputation. Our overall incentivisation framework is illustrated in Figure 2, guided by intelligent software agents, namely assessor agents (acting on behalf of service clients) and provider agents (acting on behalf of

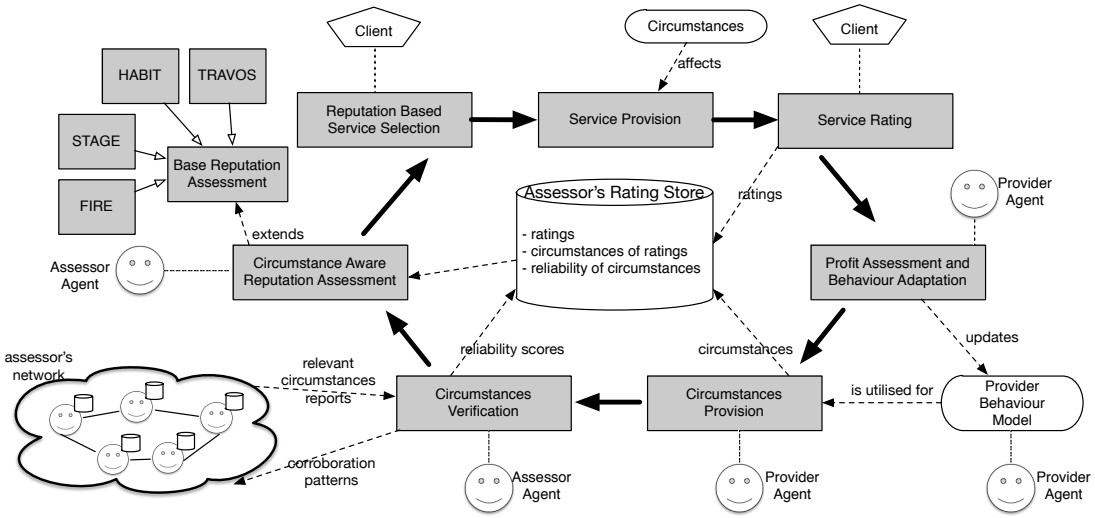
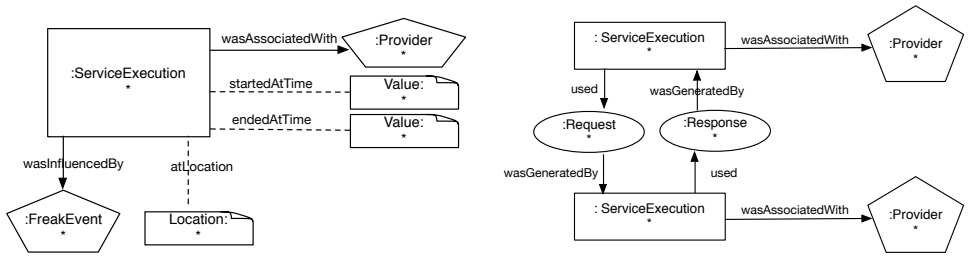


Fig. 2. Overall Incentivisation Process

service providers), which we henceforth simply refer to as assessors and providers. In particular, assessors estimate the reputation scores of providers (in *Circumstance Aware Reputation Assessment*), reflecting in these scores the availability of truthful circumstances information from providers. Guided by the reputation scores estimated for providers, clients select their interaction partners (in *Reputation Based Provider Selection*), and subsequently rate their interactions (in *Service Rating*) after service provision by the selected providers (in *Service Provision*). Client ratings are recorded in the rating stores of assessors.

Following interactions with clients, providers assess the profit achieved, and adjust their behaviour regarding circumstances provision (i.e. their *Circumstances Provision Model*) in the direction that is likely to increase their reputation, and consequently their profit, in future (in *Profit Assessment and Behaviour Adaptation*). Based on the adjusted behaviour, the provider may or may not choose to complement client ratings of its service with information of the circumstances under which service provision occurred (in *Circumstances Provision*), which are also documented in the rating stores of assessors. Finally, assessors assign reliability scores to the circumstances supplied by providers, detecting suspicious reports by confirming them against those provided by others in the population (in *Circumstances Verification*). Newly recorded client ratings, along with provider circumstances and their estimated reliability scores, are utilised by assessors in future reputation assessments of providers.

To realise the framework above, the following key components need to be specified (see Figure 2): *Circumstances Provision*: a uniform format for specifying (by the provider) and respectively interpreting (by the assessor) circumstances information; *Circumstances Verification*: a verification mechanism (by the assessor) for confirming the truthfulness of circumstances information from providers; *Circumstance Aware Reputation Assessment*: a reputation assessment mechanism (by the assessor) that accounts for the availability and truthfulness of circumstances information; *Provider Behaviour Model*: a behaviour model (by the provider) that governs the provision of circumstances information; and *Profit Assessment and Behaviour Adaptation*: a learning mechanism (by the provider) that enables the adaptation of the circumstances provision behaviour with respect to the profit received from the environment.



(a) Freak Event Pattern (b) Sub Delegation Pattern

Fig. 3. Circumstance Patterns

4.1 Circumstances Provision

Based on the patterns described in [34], we can distinguish three *types* of circumstances that may affect the provision of a service by a provider, namely occurrence of freak events $FreakEvent$, sub delegation $SubDel$, and organisation culture $OrgCul$ (affiliation with a particular organisation): $C = \{FreakEvent, SubDel, OrgCul\}$, where C is the set of circumstance types. This set is not intended to be exhaustive, but illustrative of the approach. In particular, a provider's service may be affected by a freak event, e.g. ash from an erupting volcano, flood, etc. Reliance on a good/poor sub-provider could also be a mitigating factor in a provider's good/poor performance for some aspect of the service. Similarly, the culture (values, principles and beliefs) of the organisation in which a provider operates may have an effect on the provider's performance, with the provider potentially operating differently under different organisations.

As stated earlier, the PROV standard provides a suitable solution for recording and interpreting information on the circumstances of various types underlying a service provision. For example, a *freak event pattern* that could be detected in PROV data is depicted in Figure 3(a), where $*$ is a generic individual. This pattern, denoted $ptrn(FreakEvent)$, captures the information that some freak event occurred during the execution of a service by a provider, within some time period at some location. Similarly, a *sub delegation pattern*, $ptrn(SubDel)$, is depicted in Figure 3(b), which captures the information that a service process by a provider has delegated part of the service to a service process of another provider, via respective sub-delegation request and response. Examples of other patterns can be found in [34].

A provider should supply a *circumstances report* following an interaction, also in the form of a PROV graph, detailing the circumstances encountered during their service provision in accordance with the pre-defined circumstance patterns $ptrn(c)$. An example circumstances report by a provider is depicted in Figure 4. It illustrates the occurrence of a freak event, a volcano, while executing a logistics process by a logistics compony. It also illustrates that the logistics company has sub-delegated the metal refinement task to a subcontractor.

A circumstances report by a provider for interaction i, is denoted $crmRpr(i)$. It extends the interaction's context assumed by the base reputation model (such as recency) with additional relevant information (see Equation 1):

$$ctx(i) \leftarrow ctx(i) \cup crmRpr(i) \tag{4}$$

4.2 Circumstances Verification

While circumstances reports are supplied by providers, the *reliabilities* of these reports are assessed by a third party (e.g. by the assessor after an interaction), via comparing them against the reports available from other providers.

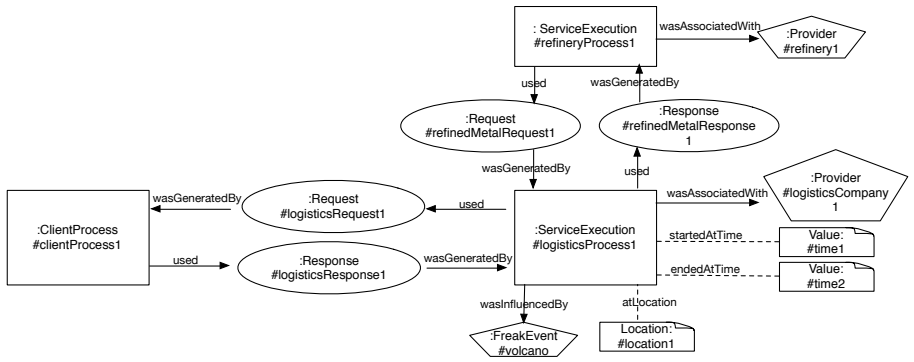
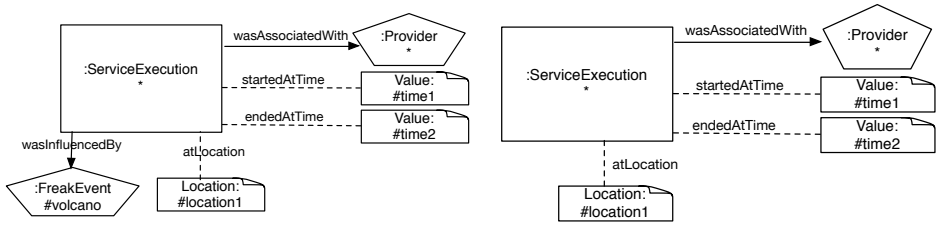


Fig. 4. A circumstances graph reporting a freak event occurrence and sub delegation by a logistics company



(a) Confirmation Pattern for the Freak Event Claim of Figure 4 (b) Witness Pattern for the Freak Event Claim of Figure 4

Fig. 5. Corroboration Patterns

In a claim by a provider regarding a circumstance type, two parts can be distinguished: a main part (which captures the core idea expressed by the claim), and a supplementary context part (which gives extra context information related to the claim). For example, a service provider may claim that its service execution, which occurred around time T at location L, was influenced by event Z. In this case, the occurrence of the event is the core idea (main part), while the time and location details are additional context information (context part). This context part indicates that the providers relevant for judging this claim's correctness are those that operated (provided services) around time T and around location L. Based on this, we can introduce the following definitions.

Definition 1. *A* confirmation pattern, $cnfPtrn(g_c, c)$, *is a query in the form of an abstracted provenance graph over circumstances reports*[2]. *It captures both the main part and the context part of a claim made by a provider regarding circumstance type* c *in circumstances report* g_c, *reflecting the information to be confirmed by the reports of others in order to assess the truthfulness of this claim.*

Example. *Consider the circumstances report* g_c *of Figure 4. The confirmation pattern,* $cnfPtrn(g_c, FreakEvent)$, *for the freak event claimed in this report is depicted in Figure 5(a). This pattern denotes a query over the circumstances reports of other providers to check whether these providers also encountered the same event (i.e. a volcano), around the same time (i.e.* $[time_1, time_2]$*), at a nearby location (i.e.* $location_1$*).*

[2]Such as query could be realised in terms of SPARQL and OWL/RDFS semantics.

Definition 2. *A* witness pattern, *$wtnPtrn(g_c, c)$, is a query in the form of an abstracted provenance graph over circumstances reports. It captures the context part of a claim made by a provider regarding circumstance type c in circumstances report g_c, characterising the witnesses relevant for assessing the truthfulness of this claim.*

Example. *The witness pattern for the freak event claimed in Figure 4 is depicted in Figure 5(b), stating that the providers relevant for judging this claim's correctness are those operated (provided services) around time frame $[time_1, time_2]$, at locations close to $location_1$.*

Definition 3. *A circumstances report (provenance graph), g_c, satisfies (implies) a pattern graph (abstracted provenance graph), g_p, denoted $g_c \Rightarrow g_p$, if there exists a projection (mapping) π from g_p to g_c, denoted $\pi(g_p)$, such that $\pi(g_p)$ is a sub-graph of g_c satisfying all the following:*

(1) *for each graph node n in g_p, $\pi(n)$ is a graph node in g_c with the same or a more restricted class, and the same individual (note that the generic individual $*$ is considered similar to any individual);*
(2) *for each connecting property pr in g_p, $\pi(pr)$ is a similar property in g_c; and*
(3) *if nodes n_1 and n_2 in g_p are connected via property pr, then $\pi(n_1)$ and $\pi(n_2)$ in g_c are connected via property $\pi(pr)$.*

Definition 4. *A* support set *of a pattern g_p, is the set of providers who supplied reports satisfying the pattern:*

$$supp(g_p) = \{p \in P \mid \exists i \in I(\_, p),\ crmRpr(i) \Rightarrow g_p\} \tag{5}$$

Based on above definitions, the *reliability* of a circumstances report $crmRpr(i)$, with respect to circumstance type $c \in C$, denoted $crmRlb(i, c) \in [0, 1]$, is computed as follows. The assessor first checks if report $crmRpr(i)$ includes a claim related to c, and if so, derives the respective corroboration patterns (witness and confirmation patterns). These patterns are used to query the provenance reports of other providers (via the assessor's network). The reliability score of the claim is then deduced from the number of supporters among relevant witnesses, as follows:

$$crmRlb(i, c) = \begin{cases} \nabla^{unc}, \text{ if } \neg(crmRpr(i) \Rightarrow ptrn(c)) \\ fn(|supp(g_m)|, |supp(g_w)|), \text{ otherwise} \end{cases} \tag{6}$$

Here, $crmRpr(i) \Rightarrow ptrn(c)$, checks if report $crmRpr(i)$ did provide information on circumstance type c, with $ptrn(c)$ being the provenance template for providing information on circumstance type c (as explained in Section 4.1). That is, $\neg(crmRpr(i) \Rightarrow ptrn(c))$ indicates that information on circumstance type c is missing (withheld) from report $crmRpr(i)$. ∇^{unc} is the uncertainty discount factor, decreasing reliability due to missing information (see Table 2 for an example instantiation). $g_m = cnfPtrn(crmRpr(i), c)$, is the confirmation pattern for the claim in report $crmRpr(i)$ regarding circumstance type c. $g_w = wtnPtrn(crmRpr(i), c)$, is the witness pattern for the claim in report $crmRpr(i)$ regarding circumstance type c. Finally fn is a function assigning higher reliability with more supporters among relevant witnesses. One example implementation of function fn, which is utilised in our later simulation, is as follows:

$$fn(|supp(g_m)|, |supp(g_w)|) = 0.9, \text{ if } \frac{|supp(g_m)|}{|supp(g_w)|} \geq 0.5; \qquad 0.1, \text{ if } \frac{|supp(g_m)|}{|supp(g_w)|} < 0.5 \tag{7}$$

In the above implementation, a report on circumstance type c is regarded as *suspicious*, and thus assigned a low reliability score of 0.1, if less than half of the relevant witnesses confirm its claim (i.e. $\frac{|supp(g_m)|}{|supp(g_w)|} < 0.5$). Otherwise, it is regarded as likely to be honest, and assigned a high reliability score of 0.9. Note that function fn, however, is not restricted to this specific implementation, and alternative implementations are also possible. For example, one area to explore in future is to not solely rely on the number of confirming witnesses, but also to account for other factors that may affect a report's reliability score such as the credibility, subjectivity and context diversity associated with

the underlying witnesses, and tuning the contributions of witnesses accordingly (e.g. by introducing weighting into Equation 7). This may involve observing the behavioural models of witnesses in order to detect subjective attitudes and fraudulent activities (e.g. as used in TRAVOS [55]), incorporating witness re-interpretation capabilities to account for subjective differences (e.g. based on Bayesian networks as in BLADE [44] and HABIT [54]), as well as ensuring that corroboration patterns (which govern who can act as a witness for a report) are rich enough to cover all relevant casual dependencies and contextual characteristics of the report (easily facilitated by the proposed provenance model).

4.3 Circumstance Aware Reputation Assessment

As discussed earlier, reputation mechanisms provide reputation scores to compare providers, estimated from ratings given to past experiences with providers. An intuitive approach would thus be to allow the circumstances given by a provider to influence the provider's reputation score via influencing the *weights* of these ratings. In fact, when circumstances are available, such an approach seems necessary to ensure accurate assessments of provider reputation for clients. In many cases, this also brings reputation benefits from the provider perspective. Consider a provider failing to deliver some goods on time on a day when an unexpected transport strike occurs. Such a failure can potentially harm the provider's reputation, but is out of the provider's control. Thus, it is advantageous for the provider to justify this failure via revealing the mitigating circumstances that occurred, to allow for its effect on reputation to be discounted.

Yet, there may be other cases where exposure of circumstances would not benefit a provider's reputation (e.g. the provider's performance is not affected by the circumstances), or might even be disadvantageous. An example of the latter case is when a particular provider delivers a poor service on a past occasion due to their reliance on a poor sub-provider. If the sub-provision circumstances were clarified, and the current circumstances are comparable (i.e. the sub-provider has not been changed to avoid such failure re-occurring), this past negative outcome would carry a greater weight on the provider's reputation, compared to the case where circumstances are not available. Moreover, such an approach (i.e. weighting past ratings by released circumstances) might also motivate providers towards the undesired behaviour of supplying untrue information. For example, a provider may claim mitigating circumstances, which did not occur, in order to justify their occasional poor performance and thus avoid reputation losses.

In order to discourage these deception opportunities (i.e. omitted or misleading information by providers), the circumstances provision behaviour of a provider should have an effect on another reputation related factor. We argue that a suitable factor is the *confidence in the weights* assigned to ratings, influencing in turn the *overall confidence* in the reputation score. The intuition behind this is as follows. When the circumstances underlying a rating are withheld, the relevance of the rating for the current situation is uncertain, which should be reflected via a low confidence in the weight assigned for the rating. Similarly, if the circumstances report is provided, but is *suspicious*, the confidence in the respectively calculated weight for the rating should also be reduced. The overall confidence in the reputation score for a provider could have an important impact on the decision making of the client, and a provider with a low confidence could potentially be placed lower in the ranking list despite having a good reputation score. For example, a provider with an average reputation score and high confidence in this score (e.g. it always releases correct circumstances reports even when these are disadvantageous for its reputation score) might be favourable to one with a slightly better reputation score but much lower confidence (e.g. it always withholds circumstances information).

Driven by above, we propose to extend the base reputation model of Equation 2 to account for provision of circumstances by providers, as follows:

$$\langle \mathcal{W}_c(a,p,i), \mathcal{WF}_c(a,p,i), \mathcal{T}_c(a,p,q), \mathcal{F}_c(a,p,q), \mathcal{S}_c(a,p,q) \rangle \tag{8}$$

where: $\mathcal{W}_c(a,p,i)$, is the circumstance-aware weighting factor, determining the relevance of interaction i for the assessment of provider p by assessor a, with consideration of extended context $ctx_c(i)$ (see Equation 4) that incorporates different types of circumstances; $\mathcal{WF}_c(a,p,i) \in [0,1]$, is the degree of confidence in weight $\mathcal{W}_c(a,p,i)$; $\mathcal{T}_c(a,p,q)$ is the circumstance-aware reputation score of provider p for term q by assessor a, accounting for weights $\mathcal{W}_c(a,p,i)$; $\mathcal{F}_c(a,p,q)$ is the overall confidence in reputation score $\mathcal{T}_c(a,p,q)$; and finally $\mathcal{S}_c(a,p,q)$ is the final score of provider p for term q upon which assessor a's decision is made, combining the reputation and confidence scores into an overall measure.

Weight $\mathcal{W}_c(a,p,i)$ is a combination of the original weighting scheme $\mathcal{W}_b$ of the base reputation mechanism (e.g. weighting by recency) and of weighting by the circumstances released by the provider:

$$\mathcal{W}_c(a,p,i) = \mathcal{W}_b(a,p,i) \times \prod_{c \in C} \mathcal{W}_c(a,p,i,c) \tag{9}$$

where $\mathcal{W}_c(a,p,i,c) \in [0,1]$ is the relevance of interaction i with respect to circumstance type c. In the case of freak events, weight $\mathcal{W}_c(a,p,i,c)$ can be computed as:

$$\mathcal{W}_c(a,p,i,FreakEvent) = \nabla^{fe}, \text{ if a freak event is claimed for } i \tag{10}$$

$$\mathcal{W}_c(a,p,i,FreakEvent) = 1, \quad \text{otherwise} \tag{11}$$

where $\nabla^{fe} \in [0,1[$ is a discount factor that decreases the relevance of the interaction given occurrence of a freak event, due to the impact of such events on service provision, which is out of the provider's control. For sub-delegation and organisation culture, weights $\mathcal{W}_c(a,p,i,c)$ correspond to the similarity degree between the circumstances under which interaction i took place and those of future interaction i_f:

$$\forall c \in \{SubDel, OrgCul\},\ \mathcal{W}_c(a,p,i,c) = sim(crmRpr(i), crmRpr(i_f), c) \tag{12}$$

where function sim returns the degree of similarity between circumstance reports $crmRpr(i)$ and $crmRpr(i_f)$ with respect to circumstance type c. Similarity between provenance graphs can be computed in a similar manner to that of conceptual graphs, utilising domain ontologies [29]. Details of this calculation is out of the scope of this paper.

Confidence degree $\mathcal{WF}_c(a,p,i)$, in interaction weight $\mathcal{W}_c(a,p,i)$, is a combination of the reliabilities of circumstances released by the provider, as follows:

$$\mathcal{WF}_c(a,p,i) = \prod_{c \in C} crmRlb(i,c) \tag{13}$$

where $crmRlb(i,c)$ is the reliability of circumstance report $crmRpr(i)$ with respect to circumstance type c (see Equation 6).

Circumstance-aware reputation score $\mathcal{T}_c(a,p,q)$, incorporates the effect of circumstances on interactions to achieve a more accurate/fair reputation assessment. It applies the same summary function of the base reputation model (i.e. as in Equation 3), but replaces the original weighting scheme $\mathcal{W}_b$, with that guided by circumstances $\mathcal{W}_c$:

$$\mathcal{T}_c(a,p,q) = \bigoplus\nolimits_{i \in I(\_,p)} \langle \mathcal{W}_c(a,p,i), r(i,q) \rangle \tag{14}$$

Circumstance-aware confidence score $\mathcal{F}_c(a,p,q)$, aggregates degrees of confidence in interaction weights, over all interactions, into an overall estimate, and combines such an estimate with the confidence score of the base reputation model to form the overall confidence score in assessed reputation:

$$\mathcal{F}_c(a,p,q) = \mathcal{F}_b(a,p,q) \times \frac{\sum_{i \in I(\_,p)} \mathcal{W}_b(a,p,i) \times \mathcal{WF}_c(a,p,i)}{\sum_{i \in I(\_,p)} \mathcal{W}_b(a,p,i)} \tag{15}$$

where the effect of $\mathcal{WF}_c(a,p,i)$ per interaction is discounted by the weight of the interaction in the base model. Note that such scaling by the original weighting scheme, $\mathcal{W}_b(a,p,i)$, of the base reputation mechanism, in Equations 9 and 15, enables tuning interaction i's relevance by other relevant factors accounted for in the base reputation model, such as recency (e.g. FIRE [23]), credibility and subjectivity of witnesses (e.g. TRAVOS [55] and STAGE [50]), and stereo type features (e.g. STAGE [50]), etc.

Finally, overall score $\mathcal{S}_c(a,p,q)$, scales the circumstance-aware reputation score of the provider according to its circumstance-aware confidence score. Note here that, whilst the reputation and confidence scores might be treated as separate measures by the base reputation model, we enforce their combination in our proposed extension for the purpose of incentivisation. In particular, $\mathcal{S}_c(a,p,q)$ is given as follows:

$$\mathcal{S}_c(a,p,q) = \mathcal{F}_c(a,p,q) \times \mathcal{T}_c(a,p,q) + (1 - \mathcal{F}_c(a,p,q)) \times \mathcal{T}_0(a,p,q) \tag{16}$$

where $\mathcal{T}_0$ is the initial default reputation assigned to a provider for a term.

From the assessor's perspective, the intuition behind the above combination is straightforward. If confidence $\mathcal{F}_c$ in available information about the provider is high, reputation $\mathcal{T}_c$, estimated based on this information, would carry a great impact on the provider's assessment, with $\mathcal{S}_c(a,p,q) = \mathcal{T}_c(a,p,q)$ when $\mathcal{F}_c(a,p,q) = 1$. In contrast, when the confidence is low, the assessor would tend to rely less on estimated reputation $\mathcal{T}_c$, assigning higher weight to the default reputation $\mathcal{T}_0$, with $\mathcal{S}_c(a,p,q) = \mathcal{T}_0(a,p,q)$ when $\mathcal{F}_c(a,p,q) = 0$.

This combination is also promising from the incentivisation perspective. For a good provider with a relatively high reputation $\mathcal{T}_c$, it would be advantageous to maintain a high confidence score $\mathcal{F}_c$ (via honest reporting); otherwise, the effect of $\mathcal{T}_c$ will be discounted by low confidence, and the provider will lose its position among competitors (those with similarly high reputation scores, but with higher confidence scores). Now, for a less good provider, with a relatively low reputation $\mathcal{T}_c$, the situation is different. Boosting the confidence score $\mathcal{F}_c$ (via honest reporting) would intensify the effect of the provider's low reputation score $\mathcal{T}_c$, and is thus not advantageous to the provider. On the other hand, lowering the confidence score with dishonest reporting is also not necessarily beneficial as this would intensify the effect of the default reputation score $\mathcal{T}_0$, which is typically low enough so that a good provider is easily favoured. Given this, it is expected that a less good provider would retreat to withholding information to escape the cost of (honest or dishonest) information provision that does bring benefits to the provider. Achieving this behaviour by a less good behaviour would still be a good result. In particular, a less good provider is not often selected due to its low score, and thus does not often provide circumstance reports. Therefore, incentivising such a provider is of less priority compared to incentivising a good provider. However, such a provider may still be occasionally selected (due to criteria other than reputation), so it is still important to ensure that it does not provide false testimonies that may jeopardise the honesty of other frequently selected providers. We further elaborate on this and demonstrate it empirically in Section 6.1.

4.4 Provider Behaviour Model and Adaptation Mechanism

The behaviour model of a provider $p \in P$, with respect to provision of circumstances, is a tuple (AC, S, Ψ). Here, AC is the set of actions available to the provider per circumstance type. In particular, we assume three possible actions per circumstance type: reporting correct information (ci), reporting false information (fi), and withholding information (wi). That is, $AC = \{ci, fi, wi\}$. S is the set of states that can be distinguished per circumstance type. We assume a circumstance type yields two possible states: occurrence of circumstances, denoted by state s^+, and no occurrence of circumstances, denoted by state s^-. That is, $S = \{s^+, s^-\}$. $\Psi : C \times S \times AC \times T \rightarrow [0,1]$, is the provider's policy function with respect to circumstances provision. It determines how the provider selects between alternative actions

for each circumstance type at each state. In particular, $\Psi(c, s, ac, t)$ corresponds to the probability of selecting action $ac \in AC$, at state $s \in S$ of circumstance type $c \in C$, at time step $t \in T$, such that: $\sum_{ac \in AC} \Psi(c, s, ac, t) = 1$. For example, $\Psi(FreakEvent, s^-, fi, t) = 0.5$ indicates that, if no freak event occurs at time step t, the probability that the provider will report false information (i.e. report an event occurrence) is 0.5. Similarly, $\Psi(FreakEvent, s^+, wi, t) = 1$ indicates that, if a freak event occurs at time step t, the provider will choose to withhold information.

Policies Ψ of the provider are not static, but change over time based on the utilities perceived by the provider from its previous actions. In particular, the provider tracks the history of its previous decisions and their respective utilities up to current time step t, $Hist\{elm(k)\}_{k=1}^{t}$, with each element $elm(k)$ of this history being a tuple,

$$\langle \hat{s}(k), \hat{ac}(k), \rho(k) \rangle \tag{17}$$

Here, $\hat{s}(k) \in \hat{S}$, is the joint state (combination of states, one per each circumstance type) that the provider encountered at time step k, and $\hat{S} = \prod_1^{|C|} S$, is the joint state space. For example, given $C = \{c_1, c_2, c_3\}$, $\hat{s}(k) = \langle s^+, s^-, s^+ \rangle$ indicates that the provider encountered circumstance types c_1 and c_3, but not c_2, at time step k. $\hat{ac}(k) \in \hat{AC}$, is the joint action (combination of actions, one per each circumstance type) that the provider performed at time step k, following the observation of joint state $\hat{s}(k)$, and $\hat{AC} = \prod_1^{|C|} AC$, is the joint action space. For example, given $C = \{c_1, c_2, c_3\}$, $\hat{ac}(k) = \langle ci, fi, ci \rangle$ indicates that the provider provided correct information regarding circumstance types c_1 and c_3, but false information on c_2, at time step k. $\rho(k) \in \mathbb{R}$, is the provider's immediate utility at time step k, specifying the immediate gain (or loss) that the provider received from the environment as a result of performing joint action $\hat{ac}(k)$ at time step k. It is determined in terms of the immediate *positive* utility $\rho^+(k)$, and immediate *negative* utility $\rho^-(k)$: $\rho(k) = \rho^+(k) - \rho^-(k)$. Specifically, $\rho^+(k)$ is the profit achieved following joint action $\hat{ac}(k)$, and is measured in terms of the number of client requests received in time interval $[k, k+1]$, multiplied by the individual profit per service provision. The latter is the income received minus the expenses incurred per provision, and is assumed to be a positive value. On the other hand, $\rho^-(k)$ corresponds to the resources consumed (e.g. time, effort, etc) by the provider to supply circumstances reports for its interactions, as entailed by $\hat{ac}(k)$. It is computed in terms of the number of interactions encountered by the provider in time interval $[k-1, k]$, multiplied by the negative cost of joint action $\hat{ac}(k)$. This cost equals to 0 if joint action $\hat{ac}(k)$ corresponds to withholding information for each circumstance type, i.e. no circumstances reports were provided.

This history of past provider actions is utilised by the provider in order to adjust its future policies in the direction that will increase its future utilities. That is, $\Psi(c, s, ac, t+1) = \textbf{learn}(Hist\{elm(k)\}_{k=1}^{t})$. An example implementation of function **learn**, which is utilised in our simulation, is to apply a form of q-learning [7], as detailed in Section 5.3.1.

5 SIMULATION MODEL

To study the effectiveness of the proposed reputation-based incentivisation approach, we conducted an agent-based simulation. The simulation proceeds in rounds (or time steps), each involving four different phases: client reconsideration; service provision; provider reconsideration; and circumstances verification. These phases are detailed below.

5.1 Client Reconsideration

In this phase, each client $a \in A$, selects an interaction partner from service providers P, for the current round. To do so, the client re-assesses the *reputation* of each candidate provider (via the respective reputation assessor), and decides accordingly. In particular, the client chooses a provider from the n most reputable ones, but also considers an exploration probability $expl_u$ where the client recursively selects a provider from the next top n ones, etc.

To determine the reputation of a provider p on behalf of the client, the assessor utilises FIRE (see Section E in the appendix) as the base reputation mechanism, augmented with the proposed extension (see Section 4.3). We also show results with BRS (see Section A in the appendix) as another example model that can be augmented with our extension, since BRS is a classic *probabilistic* reputation model that is widely cited and forms the basis for many other models (e.g. TRAVOS [55] and STAGE [55]). Our implementation assumes that all previous interactions with provider p are accessible to any reputation assessor (e.g. a fully connected client network), assigning equal importance to individual and witness experience. Note that FIRE and BRS are chosen as illustrative reputation models, and other models can be applied instead (see the Appendix for discussion of how other models can be mapped to our abstraction and so can be augmented with the extension proposed in Section 4.3).

5.2 Service Provision

In this phase, each client receives a service from the selected provider, and rates this service according to their satisfaction (which corresponds to the service level received). Each provider is assigned a specific competence level at the beginning of the simulation, but may deliver another level when subjected to particular circumstances. In this simulation, we consider *freak events* as potential circumstances affecting provision of services, as a result of which providers deliver their services at lower levels. To generate freak events, we assume a number ($routeNum$) of different service provision routes that could be followed by providers in the population, and randomly subject one of these routes to a freak event at each round. Each provider is also randomly assigned to a provision route at each round, i.e. each provider has a $routeNum^{-1}$ probability of being affected by a freak event at each round.

5.3 Provider Reconsideration

Proposed incentivisation mechanisms are commonly associated with a study of their robustness against various attacks (non-equilibrium strategies), either via a theoretical analysis [61], or via a simulation framework (empirical analysis) [28, 30]. We adopt the latter in this paper. In particular, we consider two types of providers with respect to provision of circumstances: a *rational* provider, which aims to maximise its expected return, and an *attacker*, which adopts a particular attack strategy. The simulation considers 6 attack strategies [22, 30], namely Random Attack, Constant Attack, Whitewashing Attack, Collusion Attack, Camouflage attack, and Sybil Attack.

5.3.1 Rational Provider. A rational provider adopts the behaviour model and learning mechanism of Section 4.4, which includes a detailed design/specification of the state and action spaces, and the immediate utility (reward), $\rho(t)$, perceived at each time step t. Since our simulation involves only one circumstance type (freak events), we henceforth refer to $\hat{AC}$ and AC, and to $\hat{S}$ and S, interchangeably, and omit circumstance type c from function definitions. The implementation adopted for function **learn** is as follows. A rational provider adapts its action policies Ψ at each round t, through a form of q-learning [7]: the provider tracks the immediate utilities from choosing different actions at different states, and modifies the action policies in the direction that improves its expected *long-term* return. In particular, at each round t, the provider observes the current state $s(t)$, and the immediate utility $\rho(t-1)$ of its previous action $ac(t-1)$ performed at state $s(t-1)$, and based on this, re-assesses its expected long-term return function $r(s, ac)$, as follows:

$$r(s(t-1), ac(t-1)) \leftarrow (1-\delta^r) r(s(t-1), ac(t-1)) + \delta^r (\rho(t-1) + \gamma \max_{ac' \in AC} r(s(t), ac')) \tag{18}$$

where $\delta^r \in [0, 1]$ is the return learning rate, and $\gamma \in [0, 1]$ is a discount factor reflecting uncertainty about future.

Table 1. Action Policies Referenced in the Simulation Model

Policy Label	Policy Definition
Policy 1 (Target Policy)	$\Psi(s, ci, t) = 1\ ;\ \Psi(s, ac, t) = 0,\ ac \in \{fi, wi\}$
Policy 2 (Random Policy)	$\forall ac \in AC,\ \Psi(s, ac, t) = \frac{1}{\lvert AC \rvert}$
Policy 3 (Opposite of Truth)	$\Psi(s, fi, t) = 1\ ;\ \Psi(s, ac, t) = 0,\ ac \in \{ci, wi\}$
Policy 4 (Always Reporting an Event)	$\Psi(s^-, fi, t) = 1; \Psi(s^+, ci, t) = 1\ ;\ \Psi(s, ac, t) = 0,$ otherwise

After this re-assessment, the provider correspondingly adjusts its action policy for state $s(t-1)$, by increasing the probability of the action with the maximum expected return in the long run, according to the policy learning rate $\delta^p \in [0, 1]$, as follows: $\forall ac \in AC$,

$$\Psi(s(t-1), ac, t) = \begin{cases} (1-\delta^p)\Psi(s(t-1), ac, t-1) + \delta^p & \text{if } ac = mra(s(t-1)) \\ (1-\delta^p)\Psi(s(t-1), ac, t-1) & \text{otherwise} \end{cases} \tag{19}$$

where $mra(s(t-1))$ gives the maximum return action at state $s(t-1)$, i.e. the action with a return of $\max_{ac' \in AC} r(s(t-1), ac')$. Now, based on the current state $s(t)$, and adjusted action policy Ψ, the provider selects an action according to probability $\Psi(s(t), ac, t)$, with some exploration $expl_a$. Observed state $s(t)$, and the selected action $ac(t)$, are recorded in the provider's history. The provider then supplies circumstances reports for each of its interactions in the current round according to the action selected., which are observed and recorded in the rating stores of respective clients.

5.3.2 Attacker. A *random* attacker reports information randomly, regardless of the actual circumstances. In particular, it fixes its action policy at each time step to Policy 2 of Table 1.

Constant and *whitewashing* attackers, on the other hand, always report the opposite of truth, assuming Policy 3 of Table 1. Unlike *constant* attackers, which maintain a fixed identity, *whitewashing* attackers reenter the provider population with a new identity every x rounds.

A *camouflage* attacker reports honestly in the first half of rounds (Policy 1 of Table 1), but changes their behaviour and reports the opposite of truth (Policy 3 of Table 1) in the second half of rounds.

We also consider two models of more strategic collusion attacks (which also include a camouflage element): a *slandering model*, in which colluding attackers (competent providers) falsely claim an event absence on routes affected by freak events to harm their competitors (competent rational providers) affected by such events; and a *self-promoting model*, in which attackers (less competent providers) falsely claim a freak event on non-affected routes to justify their poor performance. In both models, the colluding attackers report maliciously on some routes, but honestly on other routes, to avoid being detected and to build up their reputation. In particular, in the slandering model, attackers report dishonestly (Policy 3 of Table 1) when operating on a freak event route and another pre-agreed route, but report honestly (Policy 1 of Table 1) when operating on one of the remaining two routes. In the self-promoting model, attackers deliver bad performance and claim occurrence of freak events to justify this on three pre-agreed routes (Policy 4 of Table 1), while attempting to deliver good performance and reporting honestly (Policy 1 of Table 1) on the remaining route.

Another type of attack is a sybil attack, in which an attacker creates multiple identities, and attempts to use these identities to harm the system (e.g., by conducting one of the attack types above). The design of our system has a degree of inherent resistance against such attacks since every witness report (for confirming circumstances) must come from a valid interaction. This requires an attacker to maintain a good reputation for *each* identity (and incur the corresponding cost of performing good services), in order for the identity to be selected for interactions by clients and subsequently

Table 2. Experimental Settings

Description	Value	Description	Value
Uncertainty discount factor, ∇^{unc}	0.5	Recency factor, λ	$\frac{-5}{\log(0.5)}$
Freak event discount factor, ∇^{fe}	0.1	Number of provision routes, $routeNum$	4
Return learning rate, δ^r	0.1	Individual profit per service provision	1
Return's future uncertainty discount factor, γ	0.01	Action cost of reporting a freak event	0.3
Policy learning rate, δ^p	0.05	Action cost of reporting an event absence	0.3
Action selection exploration rate, $expl_a$	0.25	Action cost of withholding information	0
Provider selection exploration rate, $expl_u$	0.25		

act as a witness for other providers' circumstances reports. Assuming unlimited resources available to an attacker, we can simulate a sybil attack in our system by setting the majority of providers to be malicious. This is reflected in our results for the above-mentioned attack types when the proportion of attackers exceed 50% of the overall population (i.e. there are fewer rational providers than malicious ones).

5.4 Circumstances Verification

In this phase, each client (via the respective reputation assessor) attempts to establish the trustworthiness of the circumstances (if any) claimed by its interaction partner (the service provider) in the current round. To do so, the assessor queries the rating stores of the client's acquaintances for circumstances reports of other providers satisfying the confirmation pattern and the witness pattern (see Section 4.2). The results are utilised to compute the reliability scores of circumstances claimed by the provider using Equation 6. The computed reliability scores are then recorded in the assessor's rating store. In environments as open as the Web, conducting such verification might face additional challenges related to enabling heterogeneous agents to interact with each other, and with other heterogenous entities (such as rating repositories), in a uniform matter. Such challenges are discussed with a potential solution in [11].

6 EXPERIMENTAL RESULTS

We first study the effectiveness of the proposed incentivisation framework in an attack-free environment (Section 6.1), and then evaluate its robustness against different attacking strategies (Section 6.2).

6.1 Effectiveness of Incentivisation

Here, the population of provider agents are solely *rational* agents, i.e. all providers follow the policy of Equation 19. The goal is to push the provider's behaviour towards providing correct information at each state, which corresponds to Policy 1 of Table 1. In the simulation, action *ci* corresponds to reporting a freak event occurrence (for state s^+, in which the provider encountered a freak event), and reporting no freak event occurrence (for state s^-, in which the provider did not encounter any event), at the respective time and location. To assess the performance of the proposed incentivisation approach with respect to this goal, we ran several experiments comprising three sets. As a baseline, the first set of experiments involve no incentivisation, where clients depend on the base reputation model, without the proposed extensions, to assess the reputation of providers. The second set of experiments utilise a solely weighting based incentivisation strategy, where the clients assess providers according to the base reputation model, augmented with the proposed circumstance-based weighting factor $\mathcal{W}_c$, but without consideration of the circumstance-aware confidence factor $\mathcal{F}_c$. That is, reputation is assessed according to Equation 14, fixing confidence score $\mathcal{F}_c$ in Equation 16 to 1. Finally, the last set of experiments incorporate both proposed incentivisation factors $\mathcal{W}_c$ and $\mathcal{F}_c$, where the clients

assess the reputation of providers according to Equation 16. The results of each experiment set are averaged over 100 simulation runs, among each provider group. We assume two groups of providers: high competence providers (which we refer to as *good* providers), delivering favourable values for the term of interest; and medium/low competence providers (which we refer to as *less good* providers), delivering average/low values for the term of interest. Each run involves 50 good service providers, 50 less good providers, and 100 clients, with a duration of 2000 rounds. Other experimental settings are detailed in Table 2.

We first analyse the results of the three experiment sets from the perspective of a good provider (utilising FIRE as the base reputation model). Figures 6(a) and 6(b) depict the evolution of action probabilities over time for states s^- and s^+, respectively, in the case of applying the base reputation model only. As expected, in both states, providers eventually tend to withhold circumstances information. This is because provision of any information (truthful or not) in this case does not have any effect on provider reputation (the base reputation model does not account for such provision), thus keeping their positive utility (number of customers) unaffected, but incurring an additional overhead (negative utility).

In the case of solely weighting based incentivisation, there is still no advantage for a good provider to report a freak event (or report its absence) in state s^-, in which the provider delivers its normal performance. In particular, it is not beneficial here to discount the impact of occurred interactions (with standard performance) via claiming a freak event, while reporting absence of freak events would have a similar effect to that of withholding information (in both cases, the weights of occurred interactions are governed by the base reputation model). This again results in pushing the probability of the withholding information action to a high level (see Figure 6(c)) to escape the cost incurred by other actions. On the other hand, in state s^+ (where a freak event is encountered), a good provider's interactions are negatively affected. When left unjustified (by following actions *wi* or fi), such below-standard interactions carry a greater weight on the provider's reputation. This lowers its ranking score compared to its competitors (which are not necessarily affected by the event), and consequently reduces its positive utility (number of clients). Therefore, an affected good provider tends to favour reporting the event (despite the cost incurred) in order to retain its position among competitors, thus increasing the probability of action *ci* in state s^+ to a mid-range level (see Figure 6(d)). It is also evident, however, that action *wi* continues to achieve some popularity. This is because, good providers affected by events still occasionally manage to be ranked highly without any justification, achieving a positive utility when following action *wi*, without the negative cost of information provision.

Finally, Figures 6(e) and 6(f) illustrate the results of the last set of experiments (incorporating circumstance-aware confidence into circumstance-aware weighting). As can be seen, this combined strategy achieves the desired behaviour for good providers, increasing the probability of action *ci* to a high level in both states s^+ and s^-. In particular, in state s^- (Figure 6(e)), withholding information is no longer the favourable action since it lowers the confidence in the provider's reputation estimate, and consequently decreases the provider's overall ranking (see Equation 16). Action fi remains not beneficial in this state, leading to favouring action *ci* (i.e. reporting freak events absence), where the boost in confidence achieved via reporting correct information (verified by comparing this information with other providers) compensates for the negative utility of information provision. The same applies for state s^+ (Figure 6(f)). Here, the decrease in confidence from choosing actions *wi* or fi further lowers the ranking (and thus the selection chances) of the provider affected by the freak event, while choosing action *ci* (i.e. reporting the freak event) provides a double benefit (discounting the impact of the interactions affected by the event on reputation, and increasing confidence).

Now we analyse the behaviour of a less good provider (again with FIRE as the base reputation model). Such a provider delivers a below-standard interaction, regardless of whether the freak event occurs or does not occur. Here, whatever the reporting behaviour of the less good provider is (honest or dishonest), it would result either in discounting the effect

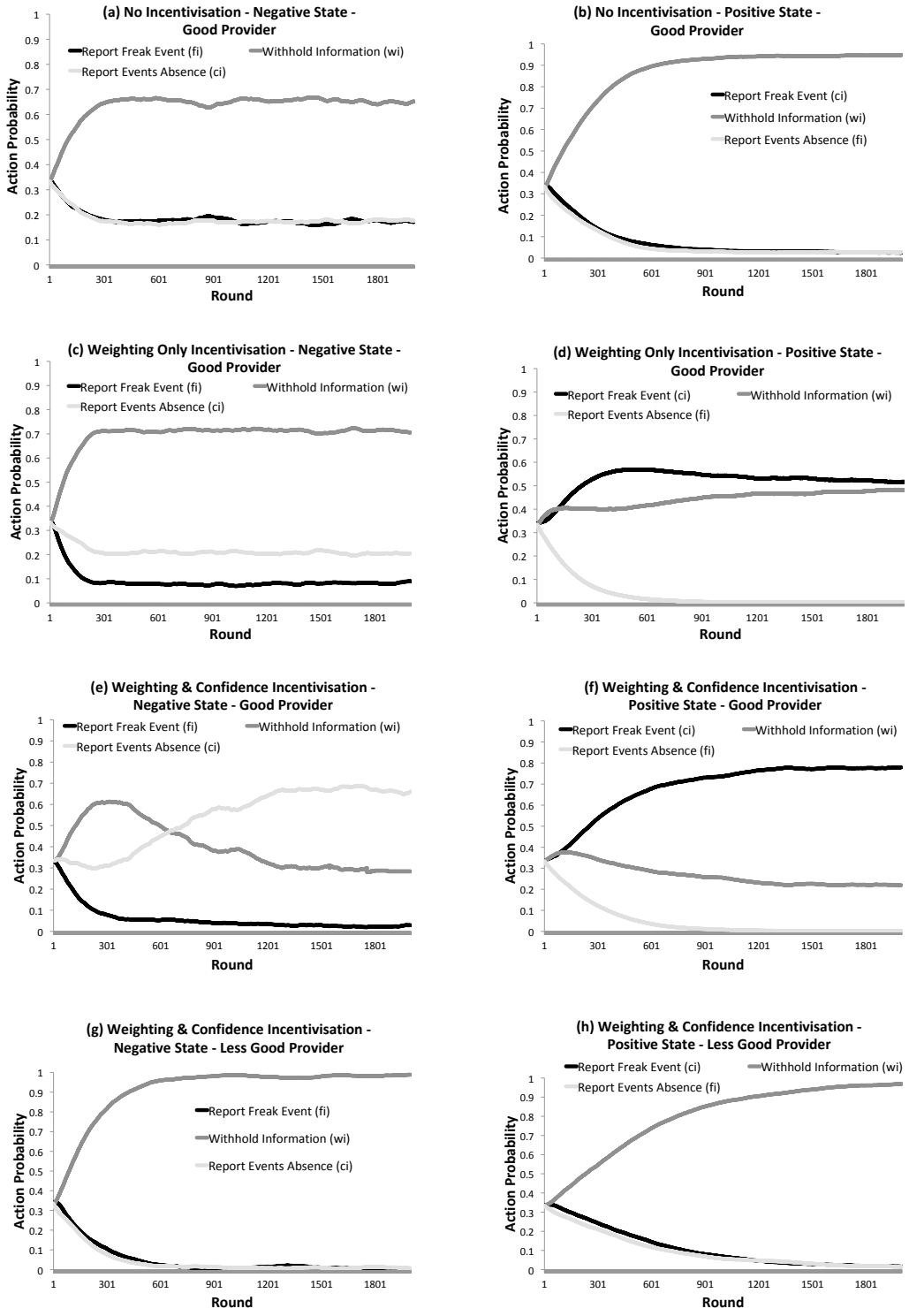


Fig. 6. Incentivisation Results for Fire as a Base Reputation Model in Attack-free Environment

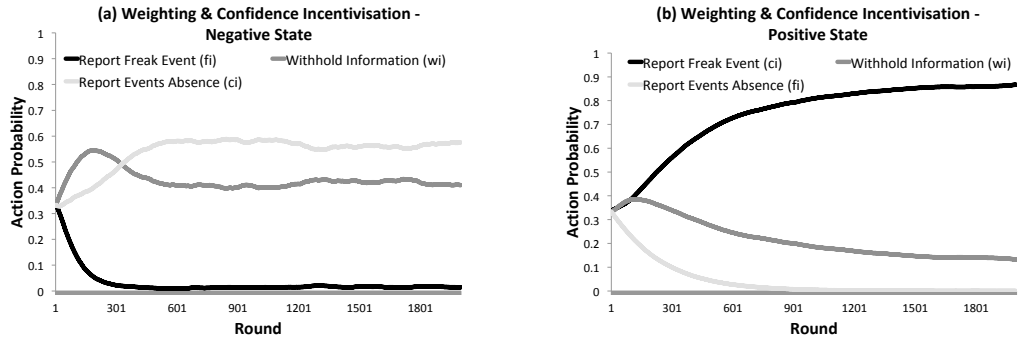


Fig. 7. Incentivisation Results for BRS as a Base Reputation Model in Attack-free Environment

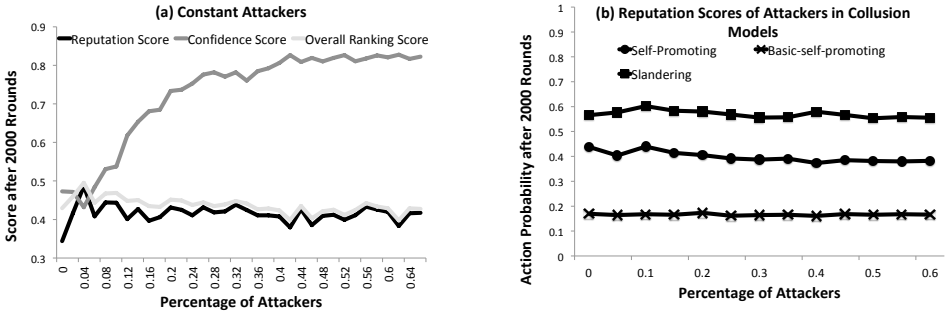


Fig. 8. Evolved Scores of Various Attackers after 2000 Rounds

of or in accounting for such bad interactions. In the former case, the provider's initial default reputation would be the one in effect. In the latter case, a bad reputation would be learned for the provider. When we set the default reputation to a low enough value (0.5 in the above experiments), a good provider is easily favoured by clients in both cases. Thus, the advantageous action for less good providers is to withhold information to escape the cost of information provision (see Figures 6(g) and 6(h)). This is still a good result as less good providers do not conduct interactions often, and so are less important to incentivise. Moreover, withholding testimonies by these providers ensures that they do not cause a negative effect on the reliability scores, and respectively on the reporting behaviour, of good providers (which are more important to incentivise).

Similar observations are achieved when utilising BRS as the base reputation model, achieving the desired behaviour for rational providers when applying the combined strategy (see Figure 7).

6.2 Robustness against Attacks

Here, we introduce attackers into the original population of rational providers. The attackers follow one of the attacking strategies outlined earlier. Unless stated otherwise, both rational providers and attackers are assumed to be *good* providers in terms of competence. FIRE is utilised as the base reputation model.

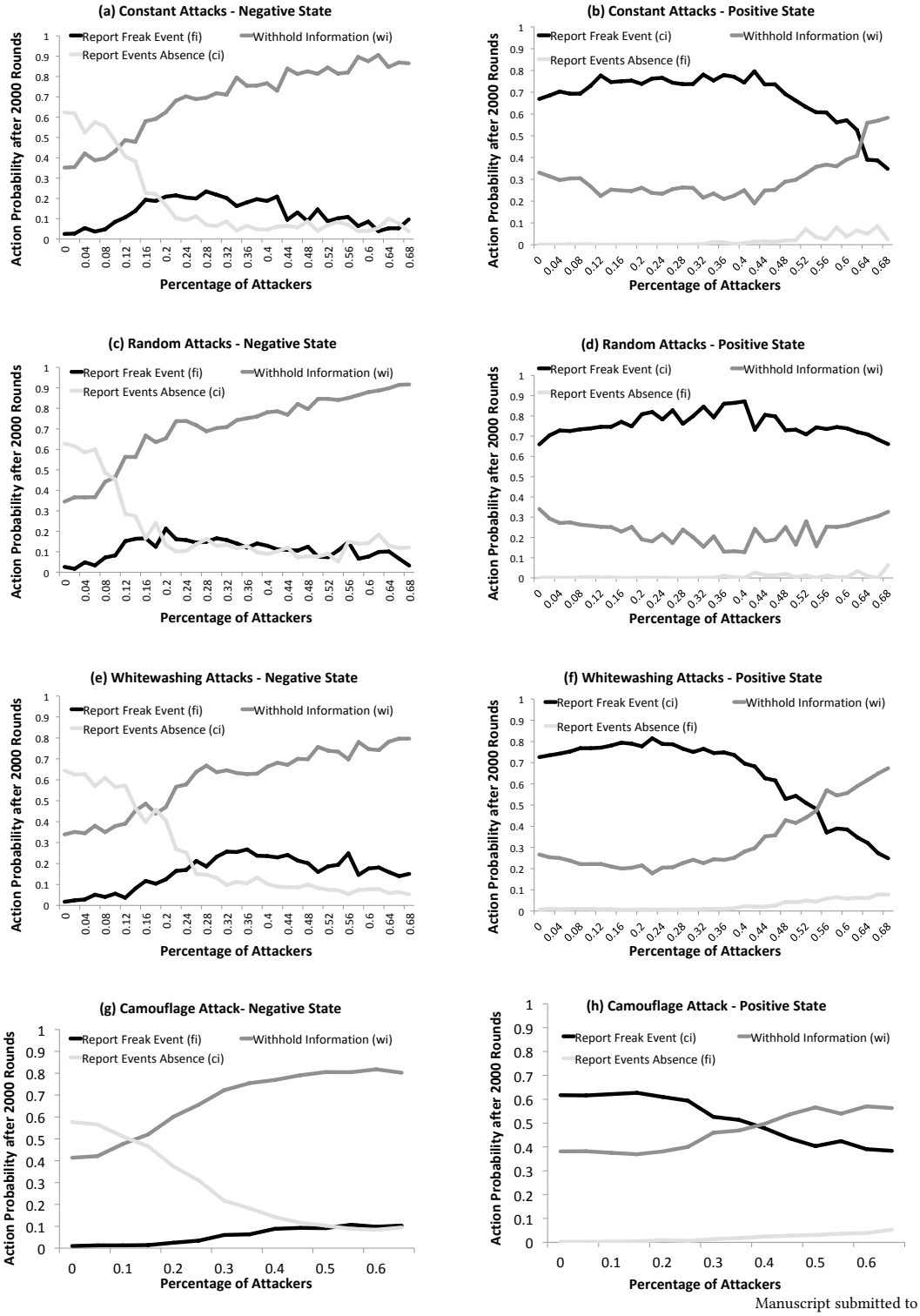


Fig. 9. Robustness Against Attacks - Evolved behaviour of Rational Providers Following 2000 Rounds

First, we study the robustness of proposed incentivisation strategy (applying both circumstance-aware weighting and circumstance-aware confidence) against *constant* attacks. Here, we assume that constant attackers constitute $a\%$ of the provider population, while $1 - a\%$ of the provider population are rational providers (which follow the learning-based policy of Equation 19). Attackers percentage $a\%$ is varied between 0% and 70%. Figures 9(a) and 9(b) show the averaged evolved behaviour of rational providers after 2000 rounds, for states s^- and s^+, respectively, whilst Figure 8(a) demonstrates the averaged evolved reputation for attackers.

From Figure 8(a), constant attackers experience a low reputation score $\mathcal{T}_c$, regardless of their percentage. This is because, when affected by freak events, the attackers suffer from poor performance, but do not justify it to clients (the attackers report a freak event absence in case of its occurrence), which negatively affects their reputation $\mathcal{T}_c$. However, attackers' confidence score $\mathcal{F}_c$, which is initially low, increases with their increasing percentage. In particular, with a low percentage of attackers, their dishonest circumstance reports are likely to deviate from those of the majority (rational providers), causing them a low confidence score $\mathcal{F}_c$, and respectively a low overall ranking score as $\mathcal{T}_0$ determines the overall score in this case (see Equation 16). Having a low ranking score (compared to rational providers), attackers would not be favoured by clients for conducting interactions, and consequently would not be allowed to act as witnesses for confirming the reports of other providers, leading to little chance for influencing rational providers. On the other hand, with a high percentage of constant attackers, the majority of the population is dishonest, supporting each other's false circumstance claims, and gaining high confidence in response. Whilst this does not boost the overall score of attackers (due to their low reputation $\mathcal{T}_c$), it pushes the confidence score of rational providers to lower levels if they report honesty. This negatively affects the latter's overall score since $\mathcal{T}_0$ starts having a greater effect on this score.

Based on this, the emergent behaviour of rational providers, with respect to different attacker percentages, can be explained as follows. In state s^+ (Figure 9(b)), the incentivisation strategy remains robust against constant attackers until their percentage reaches 64%, after which rational providers start favouring action wi (withholding information). This is because, with lower attackers percentage ($< 64\%$), action ci (reporting a freak event) continues to achieve a double benefit for rational providers (boosting both their reputation and confidence scores), thus increasing their selection chances. This is especially evident as the percentage of attackers slightly increases initially (but where they still suffer from lower ranking scores than those of rational providers), creating a less competitive environment for rational providers. However, as the percentage of attackers further increases, so does their negative effect on the confidence scores of honest rational providers (for the reasons outlined earlier). As a result, the loss in confidence score eventually cancels the gain in reputation score by action ci, leading to the rational provider favouring action wi, which does not incur an information provision cost. In state s^- (Figure 9(a)), however, robustness to constant attackers is lower, and rational providers start abandoning action ci once the percentage of attackers increases beyond 20%. Such a higher sensitivity at this state is due to relying here solely on the confidence score to incentivise action ci (reporting an event absence). As such confidence starts decreasing with the increasing percentage of attackers, honest reporting starts boosting the effect of $\mathcal{T}_0$ on the overall score, and is thus not beneficial. Some rational providers try retreating to the opposite action fi (claiming an event like the attackers) to boost their confidence score, which results in discarding their good interactions from reputation assessment. This negatively affects their reputation score $\mathcal{T}_c$, especially as they abandon justifying their bad interactions in state s^+, with the increasing attackers percentage. As a result, a rational provider eventually retreats to withholding information to escape the cost of (honest or dishonest) information provision that does bring significant benefits to the provider.

Next, we study the robustness against *random* attacks, with Figures 9(c) and 9(d) showing the respective results. Robustness is higher in the case of random attacks (compared to constant attacks), which is especially evident at state s^+,

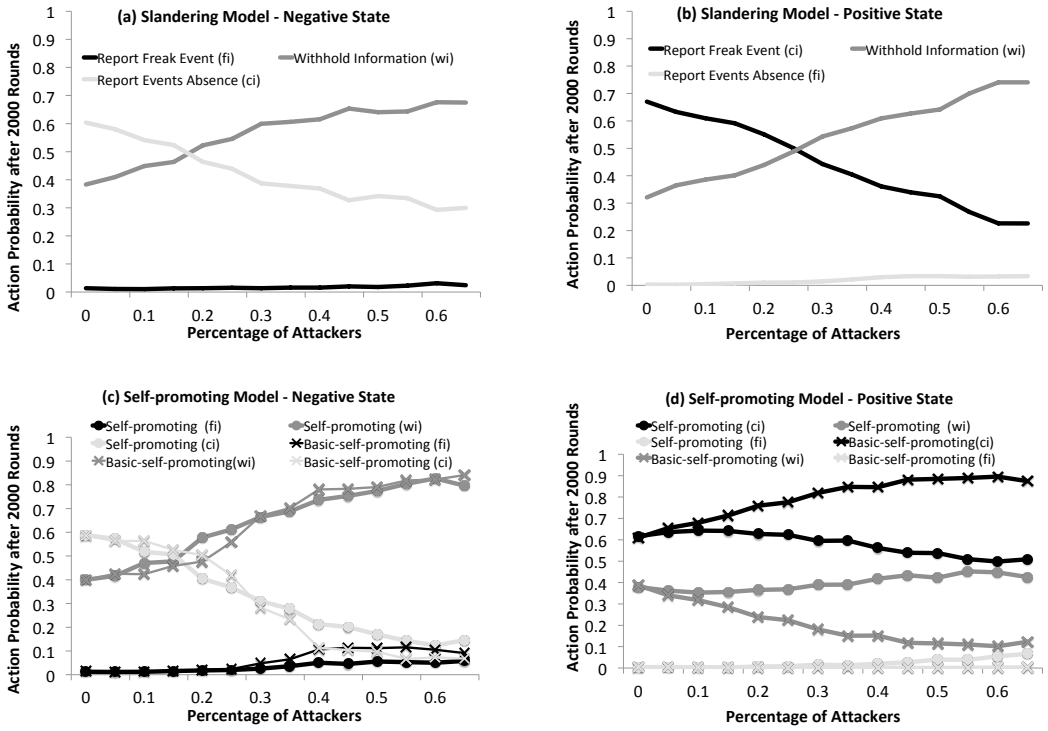


Fig. 10. Robustness Against Collusion Attacks - Evolved behaviour of Rational Providers Following 2000 Rounds

where action *ci*, despite decreasing in popularity, remains favourable over other actions with high attacker percentages. This is because, when acting as witnesses, random attackers may sometimes support truthful claims, or withhold their testimonies, thus having less negative impact on the confidence scores of honest rational providers.

Robustness to *whitewashing* attacks is studied in Figures 9(e) and 9(f), where constant attackers re-enter the population of providers with a new identity after 50 rounds. On such re-entry, attacker ranking scores are reset to 0.5 (the initial value). Similar trends to the case of constant attacks are observed here. This is because attackers generally maintain mid-range ranking scores during a run (see Figure 8(a)). Thus, restarting such scores periodically within the run to their initial (similar in this case) values by changing identity would not add much benefit to attackers, keeping their selection chances, and respectively their influence on the confidence scores of rational providers, not affected.

Robustness against *Camouflage* attacks is studied in Figures 9(g) and 9(h). As can be seen, robustness here slightly decreases compared to the case of constant attacks. This is because, attackers build high reputation and ranking scores initially via delivering good interactions and reporting honestly, which results in increased selection chances by clients. Yet, the reputation model is still able to cope quickly with attackers' changing behaviour in the second half of rounds via the recency factor, giving more importance to recent observations. This results in returning the attackers' ranking scores to mid-range values (as they suffer from poor performance due to freak events but claim their absence), which in turn reduces their selection chances and their influence on rational providers.

Robustness against *collusion* attacks (slandering model) is studied in Figures 10(a) and 10(b). Compared to the case of constant attacks, robustness here decreases in state s^+, where rational providers start favouring action *wi* (withholding information) over reporting an event after attackers percentage reaches 30%. This is because attackers here, by delivering good interactions and reporting honestly on some unaffected routes, manage to maintain a good enough reputation score (see Figure 8(b)) to be selected more often by clients for conducting interactions. This consequently enables them to act more often as witnesses for confirming the reports of rational providers, causing more harm to the latter's confidence scores in state s^+, and thus an earlier abandonment of action *ci*. On the other hand, robustness slightly increases in state s^-, as attackers support rational providers' honest claims on some unaffected routes.

Robustness against *collusion* attacks (self-promoting model) is studied in Figures 10(c) and 10(d). The figures further compare this model against a simpler form, referred to as *basic-self-promoting* model, where attackers always deliver bad performance and claim freak events to falsely justify this. As can be seen, in the basic-self-promoting model, attackers suffer from very low reputation scores $\mathcal{T}_c$ (see Figure 8(b)) despite discounting effects of bad interactions (via false reporting), as all their interactions are bad in this case. This results in very small selection chances by clients and consequently in a very small effect on the confidence scores of rational providers. The self-promoting model, on the other hand, allows attackers to raise their reputation scores $\mathcal{T}_c$ (see Figure 8(b)), improving their chances of conducting interactions and thus acting as witnesses for confirming rational provide reports. However, this increase in reputation remains insufficient to compete with the scores of rational providers, which remain favoured for interactions by clients, resulting in higher robustness to attacks (compared to the slandering model).

7 CONCLUSION

This paper presented a reputation-based framework, motivating service providers to release the circumstances in which their services were provided. Specifically, the proposed framework augments a reputation-driven system, where clients assess providers based on reputation, with the following: a verification mechanism for an assessor to confirm the truthfulness of circumstances information from providers; a learning mechanism for a provider to adapt its circumstances provision behaviour with respect to perceived reputation (in terms of the number of client requests received); and extensions to existing reputation mechanisms for an assessor to account for the availability and truthfulness of circumstances information from providers, including two factors: circumstance-aware weighting and circumstance-aware confidence. Circumstance-aware weighting (the first reputation related incentive) ensures fairness for providers by discounting the impact of poor interactions that were out of the provider control (were delivered under freak events), while circumstance-aware confidence (the second reputation related incentive) further supports this by boosting the confidence in provider reputation if true circumstances of the ratings (upon which the reputation is estimated) were made available by the provider.

Experimental results show effectiveness of the proposed framework. In particular, in an attack-free environment (i.e. with only rational providers), combining both incentives achieves the desired provider behaviour of reporting correct circumstances in all states. The framework also showed robustness towards various attack types, including constant attacks, random attacks, and whitewashing attacks. However, such robustness decreases with increasing percentage of attackers, especially in negative states where providers need to report absence of circumstances. This is because only the confidence-based incentive is relevant at these states, which is largely affected by percentage of attackers.

Whilst our analysis accounts for positive and negative utilities from the perspective of the provider (the target of incentivisation), it assumes no restriction on resources from the perspective of the assessor (the enforcer of incentivisation). That is, it currently ignores the overhead incurred by incentivisation on the enforcer side, e.g. the extra cost of

comparing the report of a provider against the reports of other providers. In future work, we plan to generalise our incentivisation framework so that it also accounts for resource boundedness and punishment costs [6] on the enforcer side. We will also explore theoretical proofs [51] to support our empirical findings.

ACKNOWLEDGMENTS

This work was part funded by the UK Engineering and Physical Sciences Research Council as part of the Justified Assessments of Service Provider Reputation project, ref. EP/M012654/1 and EP/M012662/1.

REFERENCES

[1] Luigi Atzori, Antonio Iera, and Giacomo Morabito. 2010. The Internet of Things: A Survey. *Computer Networks* (10 2010), 2787–2805.

[2] L Barakat, P Taylor, N Griffiths, and S Miles. 2017. Corroboration via Provenance Patterns. In *9th USENIX Workshop on the Theory and Practice of Provenance (TaPP 2017)*. USENIX Association.

[3] Oumaima Ben Abderrahim, Mohamed Houcine Elhdhili, and Leila Saidane. 2017. TMCoI-SIOT: A trust management system based on communities of interest for the social Internet of Things. In *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*. 747–752.

[4] Oumaima Ben Abderrahim, Mohamed Houcine Elhedhili, and Leila Saidane. 2017. CTMS-SIOT: A context-based trust management system for the social Internet of Things. In *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*. 1903–1908.

[5] Yosra Ben Saied, Alexis Olivereau, Djamal Zeghlache, and Maryline Laurent. 2013. Trust management system design for the Internet of Things: A context-aware and multi-service approach. *Computers & Security* 39 (2013), 351–365. https://doi.org/10.1016/j.cose.2013.09.001

[6] Jeremiah Blocki, Nicolas Christin, Anupam Datta, Ariel D. Procaccia, and Arunesh Sinha. 2013. Audit Games. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence* (Beijing, China) *(IJCAI '13)*. AAAI Press, 41–47.

[7] M. Bowling and M. Veloso. 2001. Rational and Convergent Learning in Stochastic Games. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence*. 1021–1026.

[8] Chris Burnett, Timothy Norman, and Katia Sycara. 2013. Stereotypical trust and bias in dynamic multiagent systems. *ACM Transactions on Intelligent Systems and Technology (TIST)* 4, 2 (2013), 26.

[9] IngRay Chen, Jia Guo, and Fenye Bao. 2016. Trust Management for SOA-Based IoT and Its Application to Service Composition. *IEEE Transactions on Services Computing* 9, 3 (2016), 482–495.

[10] Juan Chen, Zhihong Tian, Xiang Cui, Lihua Yin, and Xianzhi Wang. 2019. Trust architecture and reputation evaluation for internet of things. *Journal of Ambient Intelligence and Humanized Computing* 10, 8 (2019), 3099–3107.

[11] Andrei Ciortea, Simon Mayer, Fabien Gandon, Olivier Boissier, Alessandro Ricci, and Antoine Zimmermann. 2019. A Decade in Hindsight: The Missing Bridge Between Multi-Agent Systems and the World Wide Web. In *18th International Conference on Autonomous Agents and Multiagent Systems*. 5.

[12] P. B. Clark and J. Q. Wilson. 1961. Incentive Systems: A Theory of Organizations. *Administrative Science Quarterly* 6, 2 (1961), pp. 129–166.

[13] Giancarlo Fortino, Lidia Fotia, Fabrizio Messina, Domenico Rosaci, and Giuseppe M. L. Sarné. 2020. Trust and Reputation in the Internet of Things: State-of-the-Art and Research Challenges. *IEEE Access* 8 (2020), 60117–60125.

[14] D. Gambetta. 1988. Can we Trust Trust?. In *Trust: Making and Breaking Cooperative Relations. Ed. by D Gambetta*. 213– 237.

[15] U. Gneezy, S. Meier, and P. Rey-Biel. 2011. When and Why Incentives (Don't) Work to Modify Behavior. *Journal of Economic Perspectives* 25, 4 (2011), 191–210.

[16] N. Griffiths and K.-M. Chao (Eds.). 2010. *Agent-Based Service-Oriented Computing*. Springer.

[17] Nathan Griffiths and Simon Miles. 2013. An Architecture for Justified Assessments of Service Provider Reputation. In *Proceedings of the 10th IEEE International Conference on e-Business Engineering*. 345–352.

[18] Dominique Guinard, Vlad Trifa, Stamatis Karnouskos, Patrik Spiess, and Domnic Savio. 2010. Interacting with the SOA-Based Internet of Things: Discovery, Query, Selection, and On-Demand Provisioning of Web Services. *IEEE Transactions on Services Computing* 3, 3 (2010), 223–235.

[19] Jia Guo, Ing-Ray Chen, and Jeffrey J.P. Tsai. 2017. A Survey of Trust Computation Models for Service Management in Internet of Things Systems. *Computer Communications*. 97, C (2017), 1–14. https://doi.org/10.1016/j.comcom.2016.10.012

[20] M. Heitz, S. König, and T. Eymann. 2010. Reputation in Multi Agent Systems and the Incentives to Provide Feedback. In *Multiagent System Technologies*. Lecture Notes in Computer Science, Vol. 6251. 40–51.

[21] D. Helbing, A. Szolnoki, M. Perc, and G. Szab. 2010. Punish, but not too hard: how costly punishment spreads in the spatial public goods game. *New Journal of Physics* 12, 8 (2010), 083005.

[22] Kevin Hoffman, David Zage, and Cristina Nita-Rotaru. 2009. A Survey of Attack and Defense Techniques for Reputation Systems. *Comput. Surveys* 42, 1 (2009), 1–31.

[23] Trung Dong Huynh, Nicholas R. Jennings, and Nigel R. Shadbolt. 2006. An integrated trust and reputation model for open multi-agent systems. *J. of Autonomous Agents and Multi-Agent Systems* 13, 2 (2006), 119–154.

[24] Valérie Issarny, Georgios Bouloukakis, Nikolaos Georgantas, and Benjamin Billet. 2016. Revisiting Service-Oriented Architecture for the IoT: A Middleware Perspective. In *Service-Oriented Computing*, Quan Z. Sheng, Eleni Stroulia, Samir Tata, and Sam Bhiri (Eds.). 3–17.

[25] A. Jøsang and R. Ismail. 2002. The Beta Reputation System. In *In Proceedings of the 15th Bled Electronic Commerce Conference*.

[26] A. Jøsang, R. Ismail, and C. Boyd. 2007. A Survey of Trust and Reputation Systems for Online Service Provision. *Decision Support Systems* 43, 2 (2007), 618–644.

[27] A. Jøsang, R. Ismail, and C. Boyd. 2007. A Survey of Trust and Reputation Systems for Online Service Provision. *Decision Support Systems* 43 (2007), 618– 644.

[28] R. Kerr and R. Cohen. 2009. Smart Cheaters Do Prosper: Defeating Trust and Reputation Systems. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems*. 993–1000.

[29] Ammar Kheirbek and Yves Chiaramella. 1995. Integrating Hypermedia and Information Retrieval with Conceptual Graphs Formalism. In *Hypertext - Information Retrieval - Multimedia: Synergieeffekte elektronischer Informationssysteme, HIM'95, Konstanz, 5-7 April 1995, Proceedings*. 47–60.

[30] Y. Liu, J. Zhang, B. An, and S. Sen. 2016. A simulation framework for measuring robustness of incentive mechanisms and its implementation in reputation systems. *Autonomous Agents and Multi-Agent Systems* 30, 4 (2016), 581–600.

[31] S. Mahmoud, J. Keppens, N. Griffiths, and M. Luck. 2012. Efficient Norm Emergence through Experiential Dynamic Punishment. In *Proceedings of the 20th European Conference on Artificial Intelligence*. IOS Press, 576–581.

[32] Z. Malik and A. Bouguettaya. 2009. RATEWeb: Reputation Assessment for Trust Establishment Among Web Services. *The VLDB Journal* 18 (2009), 885–911.

[33] E. M. Maximilien and M. P. Singh. 2005. Agent-based Trust Model Involving Multiple Qualities. In *4th Int. Joint Conf. on Autonomous Agents and Multiagent Systems*. 519–526.

[34] Simon Miles and Nathan Griffiths. 2015. Incorporating Mitigating Circumstances into Reputation Assessment. In *Advances in Social Computing and Multiagent Systems - 6th International Workshop on Collaborative Agents Research and Development, CARE 2015 and Second International Workshop on Multiagent Foundations of Social Computing, MFSC 2015, Istanbul, Turkey, May 4, 2015, Revised Selected Papers*. 77–93.

[35] Luc Moreau. 2010. The Foundations for Provenance on the Web. *Foundations and Trends in Web Science* 2, 2–3 (Nov. 2010), 99–241.

[36] Luc Moreau, Ben Clifford, Juliana Freire, Joe Futrelle, Yolanda Gil, Paul Groth, Natalia Kwasnikowska, Simon Miles, Paolo Missier, Jim Myers, Beth Plale, Yogesh Simmhan, Eric Stephan, and Jan Van den Bussche. 2011. The Open Provenance Model core specification (v1.1). *Future Generation Computer Systems* 27, 6 (June 2011), 743–756.

[37] Luc Moreau, Paul Groth, James Cheney, Tim Lebo, and Simon Miles. 2015. The Rationale of PROV. *Journal of Web Semantics* (2015).

[38] Suneth Namal, Hasindu Gamaarachchi, Gyu MyoungLee, and Tai-Won Um. 2015. Autonomic trust management in cloud-based and highly dynamic IoT applications. In *2015 ITU Kaleidoscope: Trust in the Information Society (K-2015)*. 1–8. https://doi.org/10.1109/Kaleidoscope.2015.7383635

[39] N. Nikiforakis. 2008. Punishment and Counter-punishment in Public Good Games: Can we Really Govern Ourselves? *Journal of Public Economics* 92 (2008), 91–112.

[40] Michele Nitti, Roberto Girau, and Luigi Atzori. 2014. Trustworthiness Management in the Social Internet of Things. *IEEE Transactions on Knowledge and Data Engineering* 26, 5 (2014), 1253–1266.

[41] Michele Nitti, Roberto Girau, Luigi Atzori, Antonio Iera, and Giacomo Morabito. 2012. A subjective model for trustworthiness evaluation in the social Internet of Things. In *2012 IEEE 23rd International Symposium on Personal, Indoor and Mobile Radio Communications - (PIMRC)*. 18–23.

[42] G. T. Papaioannou and D. G. Stamoulis. 2010. A mechanism that provides incentives for truthful feedback in peer-to-peer systems. *Electronic Commerce Research* 10, 3 (2010), 331–362.

[43] I. Pinyol and J. Sabater-Mir. 2013. Computational trust and reputation models for open multi-agent systems: a review. *Artificial Intelligence Review* 40 (2013), 1–25.

[44] Kevin Regan, Pascal Poupart, and Robin Cohen. 2006. Bayesian Reputation Modeling in E-marketplaces Sensitive to Subjecthity, Deception and Change. In *Proceedings of the 21st National Conference on Artificial Intelligence - Volume 2* (Boston, Massachusetts). 1206–1212.

[45] M. Rodrigues and M. Luck. 2007. Cooperative Interactions: An Exchange Values Model. In *Coordination, Organizations, Institutions, and Norms in Agent Systems II*. Lecture Notes in Computer Science, Vol. 4386. 356–371.

[46] Jordi Sabater. 2004. Evaluating the ReGreT system. *Applied Artificial Intelligence* 18, 9-10 (2004), 797–813.

[47] Jordi Sabater and Carles Sierra. 2005. Review on computational trust and reputation models. *Artificial Intelligence Review* 24, 1 (2005), 33–60.

[48] Jordi Sabater-Mir and Carles Sierra. 2001. REGRET: A reputation model in gregarious societies. In *Proc. of the 4th Workshop on Deception, Fraud and Trust in Agent Societies*. 61–69.

[49] B. T. R. Savarimuthu, M. Purvis, M. Purvis, and S. Cranefield. 2009. Social norm emergence in virtual agent societies. In *Declarative Agent Languages and Technologies VI (Lecture Notes in Computer Science, Vol. 5397)*. 18–28.

[50] Murat Sensoy, Burcu Yilmaz, and Timothy J. Norman. 2016. Stage: Stereotypical Trust Assessment Through Graph Extraction. *Computational Intelligence* 32, 1 (2016), 72–101. https://doi.org/10.1111/coin.12046

[51] Victor Shnayder, Arpit Agarwal, Rafael Frongillo, and David C. Parkes. 2016. Informed Truthfulness in Multi-Task Peer Prediction. In *Proceedings of the 2016 ACM Conference on Economics and Computation* (Maastricht, The Netherlands) *(EC '16)*. ACM, New York, NY, USA, 179–196.

[52] Y. Simmhan, B. Plale, and D. Gannon. 2005. A Survey of Data Provenance in e-Science. *SIGMOD Record* 34, 3 (2005), 31–36.

[53] Dhananjay Singh, Gaurav Tripathi, and Antonio J. Jara. 2014. A survey of Internet-of-Things: Future vision, architecture, challenges and services. In *2014 IEEE World Forum on Internet of Things (WF-IoT)*. 287–292. https://doi.org/10.1109/WF-IoT.2014.6803174

[54] W. T. Luke Teacy, Michael Luck, Alex Rogers, and Nicholas R. Jennings. 2012. An Efficient and Versatile Approach to Trust and Reputation using Hierarchical Bayesian Modelling. *Artificial Intelligence* 193 (2012), 149–185.
[55] W. T. L. Teacy, J. Patel, N. R. Jennings, and M. Luck. 2005. Coping with inaccurate reputation sources: Experimental analysis of a probabilistic trust model. In *Proc. of the 4th Int. Conf. on Autonomous Agents and Multiagent Systems*. 997–1004.
[56] Thiago Teixeira, Sara Hachem, Valérie Issarny, and Nikolaos Georgantas. 2011. Service Oriented Middleware for the Internet of Things: A Perspective. In *Towards a Service-Based Internet*, Witold Abramowicz, Ignacio M. Llorente, Mike Surridge, Andrea Zisman, and Julien Vayssière (Eds.). 220–229.
[57] D. Villatoro, G. Andrighetto, J. Sabater-Mir, and R. Conte. 2011. Dynamic Sanctioning for Robust and Cost-Efficient Norm Compliance. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Spain, 16-22 July 2011.*
[58] W3C. 2013. PROV Model Primer. http://www.w3.org/TR/prov-primer/.
[59] Y. Wang and M.P. Singh. 2007. Formal Trust Model for Multiagent Systems. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*. 1551–1556.
[60] Andrew Whitby, Audun Jøsang, and Jadwiga Indulska. 2004. Filtering out unfair ratings in bayesian reputation systems. In *Proc. 7th Int. Workshop on Trust in Agent Societies*, Vol. 6. 106–117.
[61] J. Witkowski, S. Seuken, and D. C. Parkes. 2011. Incentive-Compatible Escrow Mechanisms. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence*. 751–757.
[62] Z. Xu, P. Martin, W. Powley, and F. Zulkernine. 2007. Reputation-Enhanced QoS-based Web Services Discovery. In *IEEE Int. Conf. on Web Services*. 249–256.
[63] J. Zhang, R. Cohen, and K. Larson. 2012. Combining Trust Modeling and Mechanism Design for promoting Honesty in E-Marketplaces. *Computational Intelligence* 28 (2012), 549–578.

A BRS INSTANTIATION

In BRS [25], an interaction with a provider is regarded as either successful or unsuccessful, $r(i,q) \in \{0,1\}$. The probability of success of an interaction between a and p is modelled as a random variable, with its distribution being modelled as a beta probability density function (beta pdf) with parameters α and β, and its expected value being as follows:

$$\mathcal{T}_{brs}(a,p,q) = \frac{\alpha}{\alpha+\beta} \tag{20}$$

where α represents the number of successful interactions and β represents the number of unsuccessful interactions. Specifically, α and β are defined as: $\alpha = 1 + \sum_{i \in I(\_,p) \wedge (r(i,q)=1)} \mathcal{W}_{brs}(a,p,i)$; $\beta = 1 + \sum_{i \in I(\_,p) \wedge (r(i,q)=0)} \mathcal{W}_{brs}(a,p,i)$. Here, $\mathcal{W}_{brs}(a,p,i)$ is a weighting function that assigns more weight to more recent interactions, and is defined as, $\mathcal{W}_{brs}(a,p,i) = \lambda_f^{\Delta t(i)}$, where λ_f is a forgetting factor between 0 and 1, and $\Delta t(i)$ is the time elapsed since interaction i occurred. A smaller value of λ_f means that older ratings have smaller weights and have less impact on the reputation score. When $\lambda_f = 1$, the weight is always 1 and the interaction ratings have equal impact on the reputation score. A measure of uncertainty is also provided which decreases as more evidence is gathered by the assessor: $U_{brs}(a,p,q) = \frac{2}{\alpha+\beta}$. Moreover, BRS also accounts for untrustworthy advisors, whereby reports given by witnesses with low reputations are given less weight than others.

Now, BRS can be rewritten in terms of our abstract base reputation model of Equation 2, as follows: $\mathcal{W}_b(a,p,i) = \mathcal{W}_{brs}(a,p,i)$. The reputation score of Equation 20 can be rewritten as a summary function over available ratings, considering weights $\mathcal{W}_b(a,p,i)$, as follows:

$$\mathcal{T}_b(a,p,q) = \frac{\sum_{i \in I(\_,p)} \mathcal{W}_b(a,p,i) \times r(i,q) \ + 1}{\sum_{i \in I(\_,p)} \mathcal{W}_b(a,p,i) \ + 2} \tag{21}$$

Confidence can be defined in terms of the uncertainty measure as, $\mathcal{F}_b(a,p,q) = 1 - U_{brs}(a,p,q) = 1 - \frac{2}{\sum_{i \in I(\_,p)} \mathcal{W}_b(a,p,i) \ +2}$

B TRAVOS INSTANTIATION

TRAVOS builds on BRS to better deal with inaccurate or unreliable witnesses, by comparing the opinions advisors have given in the past subsequent interactions to determine their accuracy. Specifically, witness reports, $I(w,p)$, are adjusted

based on this perceived accuracy of the witness, w, before being accounted for in Equation 20. A witness with a lower accuracy has their reports discounted more heavily than a witness who previously provided accurate reports. After the adjustment, we refer to the reports as I'. Further details can be found in [55]. TRAVOS also introduces a notion of confidence in reputation score, computed as the integral of the beta pdf over a range surrounding the expected value,

$$\mathcal{F}_{trv}(a,p,q) = \frac{\int_{\mathcal{T}_{trv}(a,p,q)-\varepsilon}^{\mathcal{T}_{trv}(a,p,q)+\varepsilon} X^{\alpha-1}(1-X)^{\beta-1}dX}{\int_0^1 U^{\alpha-1}(1-U)^{\beta-1}dU}. \tag{22}$$

Finally, TRAVOS does not weight interactions for context or recency, but instead ignores ratings that are older than a threshold λ_{trv}. Based on this, TRAVOS can be rewritten in terms of our abstract base reputation model of Equation 2, as follows. Weights $\mathcal{W}_b(a,p,i)$ in TRAVOS can be modelled as

$$\mathcal{W}_b(a,p,i) = \begin{cases} 1, & if\ \Delta t(i) < \lambda_{trv} \\ 0, & otherwise, \end{cases} \tag{23}$$

where λ_{trv} is the threshold for ignoring interaction ratings. The reputation score in TRAVOS is the same as in Equation 21 with the appropriate substitutions,

$$\mathcal{T}_b(a,p,q) = \frac{\sum_{i \in I'(\_,p)} \mathcal{W}_b(a,p,i) \times r(i,q)\ +1}{\sum_{i \in I'(\_,p)} \mathcal{W}_b(a,p,i)\ +2}. \tag{24}$$

Confidence is as given in Equation 22, i.e. $\mathcal{F}_b(a,p,q) = \mathcal{F}_{trv}(a,p,q)$.

C STAGE INSTANTIATION

STAGE [50] builds on a stereotype model introduced by Burnett et al. [8], which is based on BRS and subjective logic. In both these stereotype models, the beta pdf is shifted based on the observed stereotypes of the provider p. In Burnett et al. [8] the stereotype features are predefined, whereas STAGE introduces a method for extracting useful stereotype features from a graph based ontology. As with TRAVOS, STAGE does not weight ratings but ignores those older than a threshold. Reputation in STAGE can be modelled in our abstraction as follows:

$$\mathcal{T}_b(a,p,q) = \frac{\sum_{i \in I(\_,p)} \mathcal{W}_b(a,p,i) \times r(i,q)\ +2st(p)}{\sum_{i \in I(\_,p)} \mathcal{W}_b(a,p,i)\ +2}, \tag{25}$$

where weights $\mathcal{W}_b(a,p,i)$ are as in Equation 23, and $st(p)$ is a stereotype-reputation model that outputs a value in the range $[0,1]$. When this stereotype-reputation function outputs a constant value of $st(\_) = 0.5$, this model is equivalent to BRS, and the Beta distribution is determined only by experiences with p. In STAGE, $st(\cdot)$, is learned from past experiences using the M5 model tree learning algorithm, to output a trust score based on observed stereotype features of the provider. When there is a small amount of experience on which to judge a provider, this stereotype reputation has more of an impact on the resulting beta pdf and the overall reputation score. As more experience is gathered, the stereotype-reputation has less effect on the overall trust value.

As with TRAVOS [55] and the report filtering used by Whitby [60], STAGE also gives lower weight to opinions provided by unreliable advisors. The weights are computed by comparing the opinions that the assessor and advisor have of common providers. STAGE also incorporates stereotype trust in this weighting, using stereotype features of advisors in place of evidence, when the assessor has too few opinions in common with the advisor.

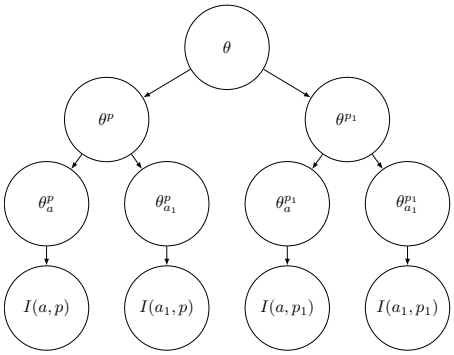


Fig. 11. HABIT reputation score function as Bayesian network.

D HABIT INSTANTIATION

Whereas BRS and TRAVOS have only one particular instantiation within our abstraction, HABIT is an abstraction itself that can be instantiated in several different ways [54]. In general, HABIT is the hierarchical Bayesian network depicted in Figure 11. In this figure, a and p are the assessor and provider agents, a_1 is an advisor or witness, and p_1 is a provider agent different to the one being assessed. If more than two providers or advisors are used in assessing reputation, their nodes are added to the network with the lattice pattern extended. For example, a third advisor would add four nodes to the Bayesian network, specifically one for $I(a_2, p)$ connected to another $\theta^{p}_{a_2}$, in turn connected to the existing θ^{p}; and $I(a_2, p_1)$ connected to $\theta^{p_1}_{a_2}$, in turn connected to θ^{p_1}. The nodes in the network can be represented by Dirichlet distributions or Gaussian distributions, of which parameters are denoted here by θ. Previous interactions between trustor agents and providers are used to parameterise these distributions, for example $I(a, p)$ are used to parametrise θ^{p}_{a}. These parameters are collated through the network using Bayesian reasoning, up to the most general parameter set, θ. The reputation score is given by the expected value of the $I(a, p)$ node, $\mathcal{T}_{hbt}(a, p, q) = E[r(i, q)|r(i, q),\ \forall i \in I(\_, \_)]$, using Bayesian reasoning of the network built from all ratings gathered, $I(\_, \_)$, including the assessor's own experience.

Using some representations, e.g. with Dirichlet processes, closed form analytical solutions exist to compute reputation score using HABIT. When analytical solutions do not exist, expected value can be computed using Monte Carlo sampling, which is computationally expensive when processing data from several advisors and with several providers.

As with BRS and TRAVOS, the interactions are not weighted in HABIT, and as such the function, $\mathcal{W}_b(a, p, i)$, is as in Equation 23. Based on this, the reputation score of HABIT can be modelled in our abstraction as

$$\mathcal{T}_b(a, p, q) = E[r(i, q)|\mathcal{W}_b(a, p, i) \times r(i, q),\ \forall i \in I(\_, \_)] \tag{26}$$

HABIT's authors note that context could be introduced by adding nodes to represent different contexts, meaning that observations from different contexts are processed as if reported by different advisors. Another approach is to introduce the weighting into the parametrisation process of the network, where ratings from contexts more similar to the current context have a higher impact on the parameter values. Confidence in HABIT again depends on the distribution being used in each of the Bayesian nodes. For Gaussian distributions, the variance of standard deviation may be appropriate, and for Dirichlet representations analytical solutions similar to that used in TRAVOS may be possible.

E FIRE INSTANTIATION

In FIRE, an interaction with a provider is assigned a rating $r(i,q) \in [-1,+1]$, where a rating of +1 is absolutely positive, −1 is absolutely negative, and 0 is neutral. The reputation of a provider from the perspective of an assessor is a combination of four components $K \in \{I, W, R, C\}$: interaction trust (I), witness reputation (W), role-based trust (R), and certified reputation (C). The last two components are not relevant for this paper, and therefore will not be considered.

The interaction trust that agent a has in p with respect to term q is determined based on a's direct experience with p, $I(a,p)$. In particular, a's past interactions with p are first scaled using an interaction weight function that gives more weight to recent interactions, as follows:

$$\mathcal{W}_{fr}(a,p,i) = recency(i) = e^{\frac{|\Delta t(i)|}{\lambda}} \tag{27}$$

where λ is the recency factor, and $\Delta t(i)$ is the time elapsed since interaction i occurred. The ratings of interactions $I(a,p)$ are then combined using a weighted mean to estimate the interaction trust, as follows:

$$\mathcal{T}_{fr,I}(a,p,q) = \frac{\sum_{i \in I(a,p)} \mathcal{W}_{fr}(a,p,i) \times r(i,q)}{\sum_{i \in I(a,p)} \mathcal{W}_{fr}(a,p,i)} \tag{28}$$

This is very similar to as in BRS, but the ratings are continuous rather than binary. Moreover, agents maintain a list of acquaintances, and use these to identify witnesses in order to evaluate witness reputation. Specifically, an evaluator a will ask its acquaintances w for their experiences $I(w,p)$ with provider p. The ratings obtained from witnesses, which are assumed to be credible, are then used to calculate witness reputation, discounting them by recency as above:

$$\mathcal{T}_{fr,W}(a,p,q) = \frac{\sum_{i \in I(w,p)} \mathcal{W}_{fr}(a,p,i) \times r(i,q)}{\sum_{i \in I(w,p)} \mathcal{W}_{fr}(a,p,i)} \tag{29}$$

The reliability of value $\mathcal{T}_{fr,K}(a,p,q)$, for each component $K \in \{I, W\}$, is $\rho_K(a,p,q)$. It is determined by a combination of the reliability of the rating set utilised (which is governed by the rating weights) and the variability of the ratings. Details of the calculations can be found in [23]. Finally, the overall reputation of a provider is calculated as a weighted mean of each of the component sources (i.e. interaction trust and witness reputation):

$$\mathcal{T}_{fr}(a,p,q) = \frac{\sum_{K \in \{I,W\}} \mathcal{W}(K) \times \mathcal{T}_{fr,K}(a,p,q)}{\sum_{K \in \{I,W\}} \mathcal{W}(K)} \tag{30}$$

where $\mathcal{W}(K) = w_K \times \rho_K(a,p,q)$, and w_I and w_W are parameters that determine the importance of each component. The reliability of the overall reputation value $\mathcal{T}_{fr}(a,p,q)$ is a combination of the reliabilities of the component sources:

$$\rho(a,p,q) = \frac{\sum_{K \in \{I,W\}} w_K \times \rho_K(a,p,q)}{\sum_{K \in \{I,W\}} w_K} \tag{31}$$

Mapping FIRE to our abstract base reputation model of Equation 2 is straightforward, as follows: FIRE's recency-based interaction weight of Equation 27 is directly mapped to the weighting function of the base model, i.e. $\mathcal{W}_b(a,p,i) = \mathcal{W}_{fr}(a,p,i)$. FIRE's overall reputation value of Equation 30 is originally expressed as a summary function over available ratings, accounting for weights $\mathcal{W}_{fr}(a,p,i)$, as implied by Equations 28 and 29, and thus can be directly mapped to the reputation function of the base model, i.e. $\mathcal{T}_b(a,p,q) = \mathcal{T}_{fr}(a,p,q)$. FIRE's reliability score for overall reputation value is directly mapped to the confidence function of base model: $\mathcal{F}_b(a,p,q) = \rho(a,p,q)$